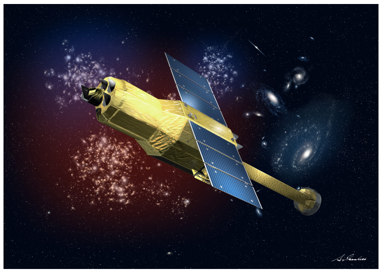


HITOMI

HITOMI DATA REDUCTION GUIDE

Version 1.0

Date: Sep 2016

X-ray Astrophysics Laboratory (NASA/Goddard Space Flight Center)
and the
Institute of Space and Astronautical Science (ISAS/JAXA)

Prepared by: Hitomi Science Data Center

CHANGE RECORD PAGE (1 of 1)

DOCUMENT TITLE: Hitomi Data Reduction Guide			
ISSUE	DATE	PAGES AFFECTED	DESCRIPTION
Version 1.0	June 24 , 2016	All	1st version
Version 1.1	Sep 20, 2016	All	Update

1 INTRODUCTION

The Hitomi mission, formerly known as ASTRO-H (Takahashi et al. 2014), is equipped with four instruments covering a wide energy range (0.3-600 keV), spanning four decades from soft X-rays to gamma-rays.

The Soft X-ray Spectrometer (SXS), which combines a lightweight Soft X-ray Telescope paired with a X-ray Calorimeter Spectrometer, provides non-dispersive 5 eV resolution spectroscopy in the 0.3-12 keV bandpass with a field of view of about 3 arcmin. The Soft X-ray Imager (SXI) expands the field of view with a new generation of 4 CCD cameras in the energy range of 0.4-12 keV at the focus of the second lightweight Soft X-ray Telescope. The Hard X-ray Imager (HXI, two units), placed on the extended optical bench, performs sensitive imaging spectroscopy in the 5-80 keV band. Lastly, the non-imaging Soft Gamma-ray Detector (SGD, two units) extends Hitomi's energy band to 600 keV. Figure 1.1 shows an exploded view of the satellite and its payload.

Figure 1.1: Schematic view of the payload and the instruments on-board Hitomi

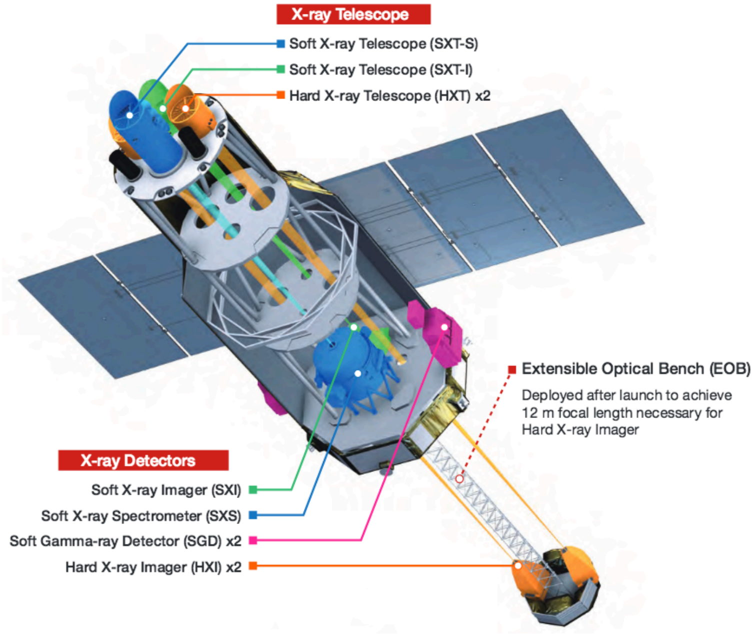
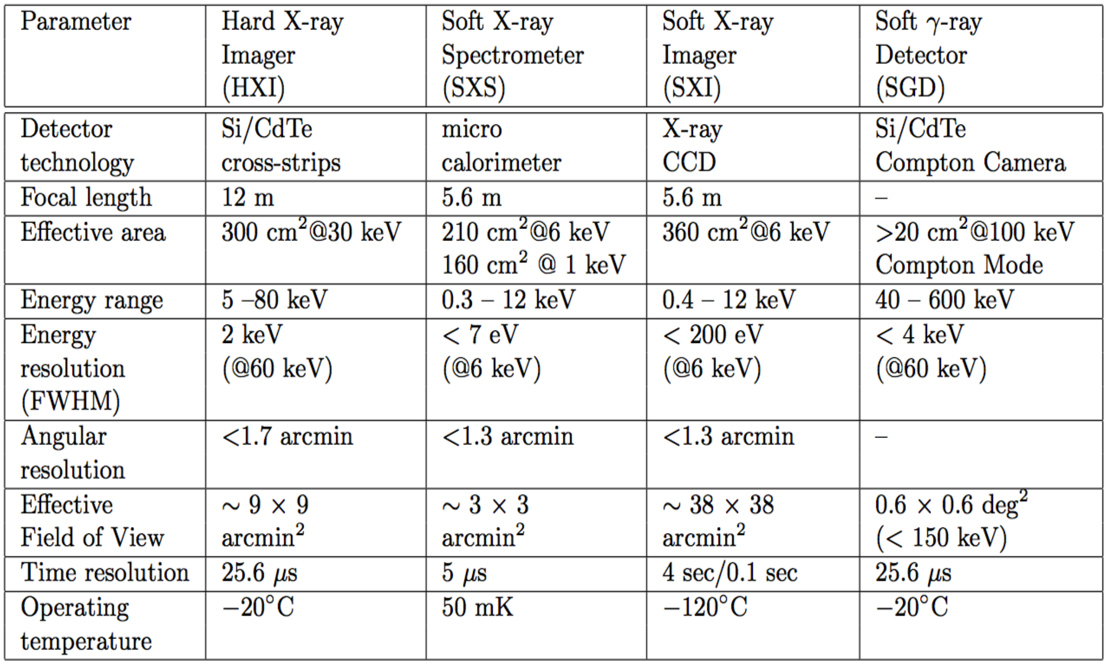

X-ray Telescope
Soft X-ray Telescope (SXT-S)
Soft X-ray Telescope (SXT-I)
Hard X-ray Telescope (HXT) x2
Extensible Optical Bench (EOB)
Deployed after launch to achieve 12 m focal length necessary for Hard X-ray Imager
X-ray Detectors
Soft X-ray Imager (SXI)
Soft X-ray Spectrometer (SXS)
Soft Gamma-ray Detector (SGD) x2
Hard X-ray Imager (HXI) x2

Table 1.1 provides an overview of the capabilities of the instruments on-board Hitomi at the time of pre-launch.



Parameter	Hard X-ray Imager (HXI)	Soft X-ray Spectrometer (SXS)	Soft X-ray Imager (SXI)	Soft γ-ray Detector (SGD)
Detector technology	Si/CdTe cross-strips	micro calorimeter	X-ray CCD	Si/CdTe Compton Camera
Focal length	12 m	5.6 m	5.6 m	–
Effective area	300 cm^2@30 keV	210 cm^2@6 keV 160 cm^2 @ 1 keV	360 cm^2@6 keV	>20 cm^2@100 keV Compton Mode
Energy range	5 –80 keV	0.3 – 12 keV	0.4 – 12 keV	40 – 600 keV
Energy resolution (FWHM)	2 keV (@60 keV)	< 7 eV (@6 keV)	< 200 eV (@6 keV)	< 4 keV (@60 keV)
Angular resolution	<1.7 arcmin	<1.3 arcmin	<1.3 arcmin	–
Effective Field of View	$\sim 9 \times 9$ arcmin^2	$\sim 3 \times 3$ arcmin^2	$\sim 38 \times 38$ arcmin^2	0.6×0.6 deg^2 (< 150 keV)
Time resolution	25.6 μs	5 μs	4 sec/0.1 sec	25.6 μs
Operating temperature	−20°C	50 mK	−120°C	−20°C

Table 1.1:Summary of the specific capabilities of the 4 Hitomi instruments (Takahashi et al. 2014).

1.1 Content of this document

This document is a guide and reference for scientists familiar with astronomical X-ray analysis and the Hitomi instruments, for learning Hitomi data analysis. General information on the Hitomi satellite may be obtained from the Hitomi Science Data Center page: http://hitomi.gsfc.nasa.gov. For help user may contact the help desk at

hitomihelp@bigbang.gsfc.nasa.gov

The document is organized as follows. Chapter 2 contains a description of the data directory structure and the nomenclature of the delivered files. It also contains the instructions to download, install, and understand the software used in Hitomi data analysis. The software and main analysis tools are explained in Chapter 3. Chapter 4 provides an overview of the data analysis, explaining in detail the pipeline and post-pipeline processing. The subsequent four chapters detail the analysis for the 4 different on-board instruments, with Chapters 5 through 8 devoted to the SXS, SXI, HXI, and SGD analysis, respectively. Chapter 9 summarizes the main analysis methods, pitfalls, and issues. The appendices contain information on the event filtering (Appendix A), a list of tasks specific to the Hitomi package (Appendix B), the list of CALDB file types (Appendix C), a summary of the fits file content for files used in the analysis (Appendix D), an explanation of the flags used in the event files (Appendix E), the sequence in which the tasks are run in the pipeline to calibrate the event data (Appendix F), and a description of the files present in the archive for each sequence (Appendix G).

2 Hitomi Data Specifics and Convention

2.1 Data

The Hitomi data are organized by sequence number and placed in the archive after processing. A sequence number may contain the entire observation on an object if shorter than one day. Observations longer than a day are divided into different sequences on the boundary between days from the start time of the observation. Within a sequence, the output of data processing includes: calibrated and screened science files, auxiliary and housekeeping (HK) files for all instruments and subsystems, as well as information related to the observation processing. All science and HK files are in FITS format. The science data follow the FITS EVENT format standard for all high energy missions. Files recording information on the processing are in plain text. The sequence number is written in the OBS_ID keyword present in every extension of the FITS files.

Each observation contains both pointing and incoming slew data for all instruments, subsystems and housekeeping. The science files for each instrument are divided using the observing mode keyword (OBS_MODE) into slew and pointing data. The instrument and spacecraft HK data are not separated by observing mode and, instead, the OBS_MODE keyword is set to 'ALL'. If an observation is divided into several sequences, the slew data are included in the first sequence number for that observation. In each sequence, the science data file may be further divided by instrument mode or filter. This information is recorded in the DATAMODE or FILTER keywords. The start and stop times of the sequence are recorded in the DATE-OBS and DATE-END keywords. All FITS files contain keywords identifying the processing version (PROCVER), the number of times this particular sequence was processed (SEQPNUM), the date of the FITS file creation (DATE), and the software and calibration version used (SOFTVER and CALDBVER). To check the file integrity, there are two additional keywords recording the data sum and checksum present in each extension (DATASUM and CHECKSUM).

2.1.1 Data Organization

All Hitomi data have unique 9-digit sequence numbers used as top-level directory names. Under these directories are a series of sub-directories, each one containing the different types of data associated with the observation. All the data files are in standard FITS format, although some output products are in Postscript, HTML, GIF, or simple ASCII. The different sub-directories are:

- auxil: Contains the auxiliary files not associated with any particular instrument. The attitude and orbit files are found under this sub-directory. Other important files include those used in the data screening, the filter (*.mkf*) and extended housekeeping *(.ehk*) files containing, respectively, instrument and satellite parameters as a function of time.
- log: Contains the log files from the pipeline processing

- hxi: Contains the data from the Hard X-ray Imager (HXI)
- sgd: Contains the data from the Soft Gamma-ray Dectector (SGD)
- sxi: Contains the data from the Soft X-ray Imager (SXI)
- sxs: Contains the data from the Soft X-ray Spectrometer (SXS)

Within each of the instrument directories, there are four subdirectories:

- hk: Contains the housekeeping data for that instrument. This includes files containing information such as voltages, temperatures, and other detector-specific data.
- event uf : Contains the Second FITS Files (SFF); these are unfiltered events files derived from the First FITS Files (FFF).
- event cl: Contains the cleaned events files. These have been filtered using the "standard" cuts on grades, SAA, etc. Data analysis initiates with these files.
- products: Contains the output products from the pipeline, such as GIF images of the data and all of the automatically-generated light curves.

The filename convention in each of these directories is instrument dependent as described below.

2.1.2 Filename convention

The output of the pipeline filename naming convention is the following:

Science and GTI file	ahXXXXXXXXXiii_PRmmmmmmmm_ll.ext.gz
Aux instrument dependent files	ahXXXXXXXXXiii_PR.ext.gz ahXXXXXXXXXiii_PR_ll.ext.gz
Aux instrument independent files	ahXXXXXXXXX_PR.ext.gz
Aux files valid for all sequence	ahXXXXXXXXX.ext.gz
Log files	ahXXXXXXXXX_nnnnnnnn.ext.gz

where

- ah: short for ASTRO-H, the old name before Hitomi
- XXXXXXXXX: 9-digit sequence number. This is also the directory name.
- iii: instrument specification:
  - 'cm1', 'cm2', 'cms' the CAMS1, CAMS2, or both CAMS, respectively;
  - 'hx1', 'hx2', 'hxi' for the HXI1 and HXI2, or both HXI, respectively;
  - 'sg1', 'sg2', 'sgd' for the SGD1 and SGD2, or both SGD, respectively;
  - 'sxi' for the SXI;
  - 'sxs' for the SXS;
  - 'gen' unless explicitly noted, for files linked to all instruments;
- P is used to identify whether the file contains data from Slew (=s), Pointing (=p), or both (=a) .

- R is a number used to separate data of the same kind if the size exceeds 2GB. The data are separated into different files; the value of R ranges from 1 to 9. If the data do not need to be separated, R=0.
- mmmmmmmm and nnnnnnn are file identifiers. This string can contain up to 8 characters and allows specific differences between files from the same instrument to be identified. The string may not include underscores or mathematical symbols.
- ll: file level. This is two-character string for FITS and GIF files.
- ext: real file extension.
- gz marks a GNU-zipped file. The file can be retrieved by issuing the command: `gunzip ahXXXXXXXXXiii_PRmmmmmmmm_ll.ext.gz`

2.2 Retrieving Data

The Hitomi data can be accessed through the HEASARC Browse interface at GSFC located at http://heasarc.gsfc.nasa.gov/cgi-bin/W3Browse/w3browse.pl (available after Oct 2016). They can also be retrieved using *wget* or FTP utilities. Both downloading methods work identically to the proprietary data access, but without the extra step of decrypting the files retrieved. In addition users can access the data at the ISAS DARTS site, although it is mainly intended for Japanese and European-based observers.

Retrieving Proprietary Data for US-based Principal Investigators

As soon as the data are processed, the Principal Investigator (PI) of the observation receives an e-mail from the Hitomi Science Data Center containing information on their FTP location and the way to access and download them. The format of the location is similar to that of ftp://legacy.gsfc.nasa.gov/hitomi/data/obs/M/NNNNNNNNN/ where M is a number indicating the type of target and NNNNNNNNN the sequence number of the data. Users are encouraged to use the command *wget* to retrieve the data:

```
wget --passive-ftp -q -nH --cut-dirs=5 -r 10 -c -N -np --retr-
symlinks ftp://legacy.gsfc.nasa.gov/hitomi/data/obs/M/NNNNNNNNN/
```

Note that the "/" at the end of the command is required.

Alternatively, this can be done using the following FTP commands:

```
ftp legacy.gsfc.nasa.gov
login: anonymous
password : your_email_address@your_domain_address
ftp> cd hitomi/data/obs/M
ftp> binary
ftp> get NNNNNNNNN.tar.gz
ftp> quit
```

Once retrieved, the data needs to be decrypted using either PGP or GPG software. General information on how to decrypt the data is available at: http://heasarc.gsfc.nasa.gov/docs/cookbook/decrypt.html.

The decryption keys for Hitomi data sometimes include special characters. Users are encouraged not to specify the key on the command line. In addition, because the *gpg* process keeps both the encrypted and decrypted versions of the files in the data directory, users are advised to check their available disk space before decrypting the data. Finally, glitches during download can prevent decryption. If an initial attempt at decrypting fails, the solution may be as simple as re-downloading the data set and trying the decryption again. *wget* is available at: http://www.gnu.org/software/wget/wget.html

2.3 Data FITS convention

All Hitomi science data files have columns that contain information on coordinates, energies, and time. Table 2.1 summarizes the FITS columns for coordinates and energy for all instruments. The timing information is written in the column TIME for all instrument and HK files.

		COORDINATES				ENERGY
		RAW /PIXEL	ACT	DET	FOC & SKY (X,Y)	PI (bin size)
		Look-down	Look-down	Look-up	Look-up	
SXS	Range	0:35	1:8 1:8	1:8 1:8	1:2430 1:2430	0.5 eV
	Pixel (arcsec)	29.982	29.982	29.982	1.768	
SXI	Range	0:639 0:319 Size for 1 segment	1:640 1:640 size for 1 CCD	1:810 1:810 Size for 4 CCD	1:2430 1:2430	6 eV
	Pixel (arcsec)	1.768	1.768	1.768	1.768	
HXI	Range	1:128 1:128	1:256 1:256	1:256 1:256	1:2430 1:2430	0.1 keV
	Pixel (arcsec)	4.297	4.297	4.297	1.768	
SGD						0.75 keV

Table 2.1: Hitomi column names for the coordinates and energy in the FITS files

2.3.1 Coordinates FITS column definitions

Hitomi has three instruments at the focal plane of the telescopes: the SXI, SXS (soft energy telescopes) and the HXI (high energy telescopes). The coordinates for these instruments are described by 5 different 'standard' coordinate systems (RAW/PIXEL, ACT, DET, FOC and SKY; see below). Since the SGDs are not imaging detectors, SGD files do not have any of the above columns for coordinates, however the XYZ position in the detector is provided for any reconstructed event. The coordinates for all instruments are defined in the teldef calibration files, one for each instrument. The SXI, SXS and HXI coordinates for the transformation can be calculated by the task ‘coordevt’, and a point-to-point or region transformation can be calculated by ‘coordpnt’ using the teldef. Figure 2.1 shows the overlay of the three imaging instruments with their references to the satellite and the sky.

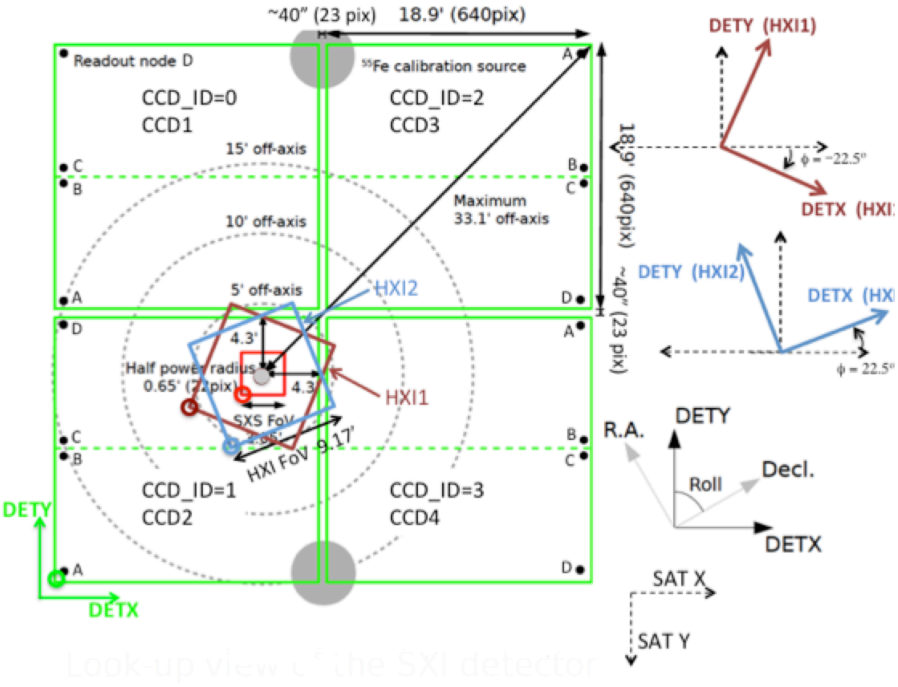


Figure 2.1: SXI look-up view of the DET system where the HXIs and SXS are overlaid. The field of view centers of the two HXI sensors and SXS coincide with the SXI pointing position. The HXI DET coordinates are also plotted (SXS is in the same sense as SXI). The origins of the coordinate system for each instrument are indicated by color-coded circles. The large grey circles represent the placement of the SXI calibration sources.

1) RAW coordinates

The “RAW” coordinates comprise the basic coordinate system. For the SXS, they correspond to the telemetered pixel number where the event landed. This value is stored in the 1-d column PIXEL and ranges from 0-35. For the SXI, the RAW coordinates correspond to the telemetered event location into a segment of the CCD. Each CCD has two segments. The RAWY ranges from 0 to 639 and RAWX from 0 to 319. For the HXI, the RAW coordinates are calculated after the event is reconstructed using the telemetered strip location. For the HXI and SXI, the RAW coordinates are defined by looking-down at the sensors.

2) ACT coordinates

The ACT coordinates are derived from the RAW coordinate system. The SXS ACT coordinates represent a 2-d look-down linearized coordinate system starting from the values stored in the PIXEL column. The SXI ACT coordinates correspond to the pixel locations in one SXI CCD. They range from 1 to 640 in the X and Y dimensions. The SXI RAW-to-ACT conversion depends on the window mode and readout node. The HXI ACT coordinates are the RAW coordinate corrected using the CAMS time-dependent misalignments.

3) DET coordinates

The DET coordinates are derived from the ACT coordinates. The SXI DET coordinates combine all four CCDs into a single system, accounting for any misalignments among them. The DET coordinates are defined as a look-up system; and therefore, for the HXI, SXI, and SXS the transformations from ACT to DET include a flip such that the satellite +Y direction becomes the -DETY direction (and the satellite +X is parallel to the +DETX (similar to the Suzaku convention).

4) FOC coordinates

The FOC system combines all of the HXI, SXI, and SXS individual DET coordinates into a common system, accounting for misalignments among them. Since the SXI has the largest field of view and smallest detector pixel size, the FOC coordinates for all sensors adopt the pixel scale and range for the SXI. Misalignments between the sensors are taken into account so that the different FOC images can be superposed. FOC is calculated from DET by linear transformations that represent the instrumental misalignments.

5) SKY coordinates

SKY coordinates populate the columns X and Y from where is it possible to calculate the RA and DEC for each pixel using the WCS keywords. The SKY system, represented in WCS coordinates, is a tangent plane projection and is oriented such that declination (δ) increases in the +Y direction and Right Ascension (α) increases in the –X direction. The conversion from FOC to XY (SKY) uses the satellite attitude, and they differ by the position angle or roll.

2.3.2 Energy FITS column definition

The HXI, SGD, SXI, and SXS spectra are derived using the FITS column PI. The PI (pulse invariant) values are the linearized pulse height (channel) where each channel has equal energy width. This is calculated with the algorithm appropriate to each of the instruments. The telemetered ‘energy’ values for all instruments are stored in the column PHA. Additional columns associated with energy are populating when calculating PI to account for the different corrections required by each instrument.

1) SXS

The SXS telemetered ‘energy’ information is stored as PHA for each event. The ‘sxssecor’ task corrects the PHA of secondary events. The corrected PHA for the secondary events, as well as the telemetered PHA for primary events, are written in the column PHA2. The PI is calculated using ‘sxsgain’ and ‘sxspha2pi’ tasks. ‘sxsgain’ calculates the SXS time-dependent energy correction using a calibration line and derives the appropriate temperature for that correction. Using the results of ‘sxsgain’ and the

SXS CALDB gain file, ‘sxspha2pi’ calculates the energy columns EPI and EPI2. The EPI values are derived using the PHA column as if all events are primary. The EPI2 is calculated using the PHA2, which contains the correction for secondary events. The PI column is calculated using EPI2. The PHA, PHA2 and PI units are channel, the EPI and EPI2 units are eV.

2) SXI

The SXI telemetered ‘energy’ is stored in the ‘PHAS’ column that contains the 3x3 pixel array telemetered charge around the event. The PI is calculated using the tasks ‘sxipi’.
The task first corrects the values in the 3x3 array for the video temperature part of the CCD electronics, the charge trail, and the charge transfer inefficiency. Using the corrected 3x3 array, ‘sxipi’ calculates the grade and the pulse height for each event and populates the GRADE and PHA columns. Lastly, the gain is applied to each event and the computed pulse invariant stored in the PI column.

3) HXI

An X-ray photon, upon interacting with the detector (hereafter, referred to as an occurrence) may create more than one signal; and, the HXI telemeters an ‘energy’ for each signal detected in the detector. The PHA column contains the telemetered ‘energy’ values and, since within an occurrence the number of signals is variable, the column has a variable length. The PI is calculated using ‘hxisgdpha’ and ‘hxievtid’ tasks. ‘hxisgdpha’ corrects the PHA for the gain for each signal, and stores the information in a variable length array column EPI (units: keV). ‘hxievtid’ determines whether the signals in an occurrence are consistent with a valid event by comparing with specific template patterns of signals in the detector. If an occurrence is consistent with an event, ‘hxievtid’ computes the event position (column RAWX and RAWY) and the PI is calculated using the EPI values.

4) SGD

The PI calculation in the SGD is done similarly to the HXI, and uses the tasks ‘hxisgdpha’ and ‘sgdevtid’. ‘hxisgdpha’ calculates the EPI column similarly to the HXI. ‘sgdevtid’ analyzes the signals in an occurrence to determine if they are consistent with a valid event. If an occurrence is consistent with an event, ‘sgdevtid’ computes the PI value.

2.3.3 Timing Information

Hitomi has a GPS receiver on board that dispatches the time to all subsystems. The instruments electronics assign the fine time to the science data. The time assignment uses the information from the GPS as well as the fine time. The task ‘ahtime’ calculates the times for all the science data as well as the HK. The times are stored in the column TIME which contains seconds since an epoch set to 2014-01-01 00:00:00 UTC. The epoch is written in the header of all FITS files as an MJD value in the keywords MJDREFI and MJDREFF. The time system is TT (Terrestrial Time) and the MJD value of the epoch in TT corresponds to MJD 56658.0007775925926 (TT).

3 SOFTWARE AND CALIBRATION

3.1 Hitomi Software Package

The Hitomi software package is developed and distributed in the HEASoft environment (http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/). HEASoft (also known as "FTOOLS") is a multi-mission collection of programs and scripts with similar interface that allows tasks to be run both interactively and in scripts. The calibration and analysis of the Hitomi data requires mission-specific tasks, as well as multi-mission tasks. The Hitomi package has been developed by the Software Calibration Team at Goddard Space Flight Center (GSFC) in collaboration with the Instrument Teams and supports all instruments and subsystem on board Hitomi. When downloading the Hitomi package from the HEASARC website, standard packages such as XANADU (*ximage, xronos* and *xspec*) for imaging, spectral and timing analysis; multi-mission tools applicable to high energy astrophysical data including *xselect* (general tool to extract the products); and other sub-packages with general fits utilities to manipulate FITS files are also included. The Hitomi package allows the users to recalibrate the data (when new calibration information is made available), filter the data, compute redistribution matrix (RMF) and auxiliary response (ARF) files that conform to the XSPEC response and ARF standard formats, and extract spectra, images or light curves as required by their scientific research. The list of tasks in the Hitomi package can be obtained by invoking "fhelp hitomi" after software installation (see also Appendix B for the list of tasks). The software operates on Hitomi FITS files present in the archive and uses CALDB to store the calibration data. The XANADU software packages guide is available from http://heasarc.gsfc.nasa.gov/docs/xanadu/xanadu.html. HEASoft (that includes the Hitomi package) is supported on major Unix architectures, such as Linux and Mac OS X. The supported operating systems are listed in the HEASoft website.

3.2 CALDB

The Hitomi calibration information is stored in the HEASARC Calibration Database (CALDB; http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/caldb_intro.html). The calibration information is stored in files and an index, one for each instrument, that records the valid files. CALDB must be installed to use the CALDB appropriate to the tasks in the Hitomi package. CALDB can be installed on local machines or accessed remotely. The latter ensures that the most up-to-date version is used, but there may be a penalty in terms of speed of access. To install the calibration file locally, users need to download the CALDB tar files from:
http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/caldb_supported_missions.html.
To set up access to the local CALDB installation, users should source the file *caldbinit.csh* or *caldbinit.sh,* depending on their favorite shell. These scripts set up the environment variables that are necessary for the use of CALDB, and may be found in the CALDB tree in the directory software/tools. Note that these scripts must be edited to point to the directory on the user system where CALDB has been installed. The remote access method is explained in more detail at http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/caldb_remote_access.html

After CALDB has been installed, the Hitomi tasks that required calibration information may be run by either specifying the paths to individual CALDB files or by simply entering CALDB. The second method is recommended, since the tasks then automatically query the CALDB index file and retrieve the most up-to-date valid calibration files.

3.3 XSELECT

After data has been calibrated (see Chapter 5 to 8), the Hitomi event files may be used within the mission independent software interface, *xselect,* to extract high level products such as images, lightcurves, or spectra. *xselect* is included in the Hitomi package. Users unfamiliar with *xselect* should read the manual available at: https://heasarc.gsfc.nasa.gov/ftools/xselect/xselect.html.
To work properly, *xselect* requires an MDB file included in the *xselect* distribution, that sets the parameters of the different on-board instruments. The file contains the standard column settings for a given mission necessary to extract spectra, light curves and, whenever possible, images from the data. The distributed MDB file is set to allow image extraction for all Hitomi instruments except the SGD and data in the SXS anti-coincidence mode. Users are advised to use *xselect* exclusively to extract products (spectra, light-curves and images.). The specific Hitomi settings in the *xselect* MDB are the following:

- The light curves are derived from the TIME column with a default binning set to 16 sec.
- The spectra for all instruments are derived from the PI column. No bad channels or binning are set for any of the instruments.
- Sky images are derived from the X/Y columns and valid only for HXI, SXI and SXS.
- Images in raw and detector coordinates use the columns ACTX/ACTY and DETX/DETY (valid only for the HXI, SXI, and SXS).
- A WMAP is created only for the SXI using the DETX/DETY coordinates, and placed in the primary header of the SXI spectral file. This is necessary to create the RMF.
- Spectra, light curves, and images are derived using the *extractor* task.
- The filter file has the name ah*.mkf*
- Event file names have the strings: ah*hxi1*, ah*hxi2, ah*sgd1*, ah*sgd2*, ah*sxi*, ah*sxs* (note for HXI and SGD, only the file type *ufa* or *cl* may be input in *xselect* and not the *uf* file).

3.4 XSLIDE

XSLIDE (X-ray Spectral Line Identification Energy) is a software package for detecting and identifying lines the energy spectra. XSLIDE uses a graphical user interface to assist users in the analysis. It requires a spectrum and an ARF with formats compatible with those used in *xspec*, and is therefore suitable for SXS spectra. XSLIDE allows for line identification, as well as basic diagnostics, on the lines and plasma using information stored in external files. The package is not currently distributed with HEASoft and the software package can be downloaded from ftp://legacy.gsfc.nasa.gov/hitomi. The tar file includes the installation guide as well as the user guide.

4 Hitomi DATA ANALYSIS OVERVIEW

4.1 Pipeline

The Hitomi pipeline consists of a collection of scripts to manage data files and process all the science and HK FITS files from all instruments and subsystems within a sequence. The input files are the first FITS file (hereafter FFF) generated from the pre-pipeline in ISAS (Japan) directly from the telemetry but for the attitude and orbit. The attitude and orbit files are created by a special task, not included in the pipeline not distributed in the Hitomi package. The FFF files have the structure and the number of columns of the final FITS files but the pre-pipeline only populates the columns with the telemetered values and as an exception the TIME column to ensure that also the attitude and orbit data are time tagged consistently.

The pipeline runs at GSFC (USA) and populates the remaining columns in the science FFF files. At the end of each run the files within a sequence are packaged and sent to the archives. The processing script used in the pipeline includes the calibration of the energy and position for each event, calculation of good time intervals from the different subsystems and HK, screening of the science data as well as the creation of supporting files to help the users in their analysis and the instrument team to monitor specific parameters.

The tasks used to process the data in the pipeline as well as to calculate the timing information used in the pre-pipeline are all part of the Hitomi package so that users may always recalibrate or rescreen the data if new calibration information is available or special screened is needed in their analysis.

4.2 Reprocessing

The pipeline outputs and the software are designed to allow user to reprocess the data, if necessary, by using scripts included in the Hitomi package. Data reprocessing is recommended if there is an update of the calibration files or a change in the Hitomi software package or in the processing script compared to what it was used in processing pipeline. The header of all data files includes a set of keywords to record the pipeline, the software and calibration versions. These are:

- PROCVER records the version of the processing script
- SOFTVER records the version of the Hitomi package used in the pipeline
- CALDBVER records the calibration release by instrument used in the pipeline

Before reprocess the data, users should check for any updates of software or calibration or processing version at https://heasarc.gsfc.nasa.gov/docs/hitomi/analysis/.

To reprocess the data the Hitomi package includes a script *'ahpipeline'* which duplicates the pipeline operations. The task operates on all instruments and is divided in three stages: 1) Calibration, 2) Data screening and 3) Product creation. It allows the user to run all or part of the pipeline processing and to vary the calibration files and the filtering or screening criteria used. *'ahpipeline'* allows the data processing from all or a sub-set of the instruments. It is quite flexible since allows to change the parameter setting for any tasks used in the pipeline. The Hitomi package also includes the tasks *'sxspipeline', 'sxipipeline', 'hxipipeline' and 'sgdpipeline'* which are essentially equivalent to run '*ahpipeline*' for only a particular set of instruments and they are directly used in *'ahpipeline'*. Here it is an example of how to run *'ahpipeline'* to reprocess and screen all data from all instruments within a sequence. This step is recommended if relevant CALDB files have been updated since the original cleaned event files were created.

```
ahpipeline indir=/path/to/12345678 outdir=/path/to/12345678_repro \
entry_stage=1 exit_stage=2 steminputs=ah123456789 instrument=ALL
```

With this command, the script automatically finds unfiltered event under the directory of each instrument, and creates reprocessed data in 'outdir'. Other examples may be found in the following instrument sections.

4.3 Data Selection and products

The event files products of the pipeline are used to extract a light curve, an image or a spectrum The output of the pipeline includes two types of event files: the 'unfiltered' file (`_uf.evt`) and the 'cleaned file (`_cl.evt`). The 'unfiltered' file has all events calibrated where no screened has been applied therefore contains all the telemetered events. The 'cleaned' file contains instead events that has been screened either for specific characteristic and/or or by special time interval has been applied. The screening criteria applied in the pipeline are 'pre-determinate' and they are listed in each of instrument chapter together with additional suggestions for further screening of the 'cleaned' event.
To extract products users may start from the cleaned event if the pre-determined screening criteria are satisfactory for their science goals else start from the unfiltered file and applied a different or modified set of screening criteria.The Hitomi package contains two tasks that allow data screening 'ahgtigen' and 'ahscreen' (see also Appendix A). 'ahgtigen' can either create a GTI file using an expression operating on columns stored in a FITS file containing a TIME column, or it can merge two or more GTI files. 'ahscreen' screens science event files by applying an expression that filters on specified columns in the event file, and/or by applying a GTI file.
After selecting the appropriate event file, user can extract products using *'xselect'* or use directly the event file in multi-mission task that allow to read directly event files (*'ximage'* and *'ds9'* for images, *'xronos'* tasks for light curves).

4.4 Data Analysis

4.4.1 Imaging Analysis

The event files for the imaging instruments SXS, SXI and HXI can be read into *xselect* or *ximage* or *ds9* to create images in different coordinates. To create an image in *xselect* the commands are:

```
  xselect> read event event_filename
  xselect >extract image
  xselect>save image image_filename
```

To create an image in *ximage* the commands are:

```
  ximage > read event_filename
  ximage > write image_filename
```

To view an image in *ds9* is sufficient to type on the command line '*ds9 event_filename'* and use the graphical user interface to save an image in a file.
These images may be used to define a sky region and to further select the events (see specific region definition to extract spectra for the different instrument).

Instrument field of view
The field of view of the HXI, SXI and SXS is very different to each other and as first step the user may want to assess which areas of the sky fall into the different detectors. The task *'ahmkregion'* allows to create field of view regions for all imaging instruments that may be used to overlay on a image. These FOV region files may contain labels to identify segment in the CCD for the SXI detector or pixel number for the SXS detector. These regions are in the standard SAO format and can be read within *ximage* or *ds9.*

Flat Field
For study of extended sources images need to be corrected for positional dependent effects due to the telescope and/or the detectors. This correction is done by created a flat-field efficiency map, an image of the same size of the sky image containing in each pixel a correction factor. There are two tasks to create a flat field : *'ahexpmap'* for the SXI and SXS and *'hxirspeffimg'* for the HXI. *'ahexpmap'* includes in the flat field for the SXI and SXS corrections due to the telescope vignetting, detector quantum efficiency, contamination, bad pixel and exposure. In the case of the SXS, there are additional effects due to the gate valve if it is closed, and any filters in the optical path that have a spatially non-uniform opacity. *'hxirspeffimg'* includes in the telescope and the positional quantum efficiency effects as a function of time using the information from the alignment system from the CAMS subsystems.
The flat-field maps are created for specific energy range. The energy range should to match the energy range of the sky image obtained from the event file.

4.4.2 Spectral Analysis

Overview
To extract a spectrum from the HXI, SXI and SXS event files, the first step is to define an extraction region. For point-like (unresolved at the Hitomi resolution) sources it is recommended to use circular extraction regions of radius 2.5 and 3 arcminutes for the SXI and HXI, respectively. The extraction region must also include any regions to be excluded (e.g., due to contaminating sources), and in the case of the SXI exclusion regions may also be made for calibration sources. Additional region files may be constructed to extract background spectra.
For the SXS, the extraction region must be made of boxes that encompass the original individual pixels of the SXS array. To omit individual pixel from the SXS spectrum it is recommended to use exclusion region and not filtering the event file using the PIXEL column. Also it is not recommended to extract spectrum from individual pixels. The SXS PSF has a complicated azimuthal structure and the SXS entire array is smaller compared to the PSF. So counts in an individual pixel (size ~30 arcsec) may vary widly with the position of a pixel on the PSF. Attitude variations may also cause large fluctuations within a pixel. The azimuthal PSF structure is not calibrated to sufficient accuracy to get accurate effective area within individual pixel.The SGD do not require an extraction region to create a spectrum.

All spectra maybe extracted within *xselect* or by specific script as noted in the following chapter dedicated to each instrument.

Once spectra are extracted, the next step is to make spectral response files that can be used with the spectral data to perform model-fitting with spectral-fitting packages such as XSPEC. RMF files are generated using the sxsmkrmf script for the SXS, and the *sxirmf* tool for the SXI. The arf are generated by *aharfgen* which runs the tool *ahsxtarfge*n for the SXS and SXI to make an ARF, and *hxirspeffimg* to make a net response matrix (RSP file) for the HXI (using RMF files in CALDB). An input required by *aharfgen* is the exposure map that is calculated by the tool *ahexpmap.* The output file from *ahexpmap* carries information about the satellite attitude variation during the observation in a form that can be used by *aharfgen*. A separate task, *sgdarfgen*, is used to construct an off-axis RSP file for the SGD (pre-made files in CALDB in the on-axis case). The HXI and SGD may be corrected for deadtime, using the pseudo events, by the tool *hxisgddtime*.

The tools to construct non-X-ray background (NXB) spectra for each instrument are *sxsnxbgen* (SXS), *sxinxbgen* (SXI), and *hxinxbgen* (HXI). NXB spectra may be subtracted from the on-source spectrum. In that case, the sky background must be modeled by including additional model components, or a separate NXB-subtracted sky spectrum may be simultaneously fit. Alternatively, an off source spectrum can be subtracted. The background may be negligible in some cases, depending on the energy band of interest and source brightness and size.

Spectral Response Imaging Instruments (SXI, SXS HXI)

The RMF is a spectral redistribution matrix (also known as the "line-spread function," or LSF), while the ARF is a multiplicative function of energy and contains all the information about the effective area (EA) due to the X-ray telescope Quantum Efficiency (QE), and a number of other detector-related efficiencies described later. The RMF and ARF are made separately for the SXS and SXI while, for the HXI units only the net response matrix (RSP) file is created due to the very different principles involved in how the HXI functions.

For the HXI the LSF and QE is different for each of the 5 layers in a single HXI unit, so the net response matrix is a weighted combination of the 5 responses. The weights depends on the relative position-dependent X-ray source counts, which in turn depend on the effective area (see also CAMS).

The *aharfgen* script makes ARF files for the SXS and SXI, and RSP files for HXI1 and HXI2.

Although the scheme for creating response functions differs, there are some procedures that all of the instruments have in common. The telescope effective area is calculated by the ray-tracing, which is essentially a Monte Carlo simulation of photons injected into the telescope aperture. The resulting response function has a statistical uncertainty (which is generally energy-dependent) that it is controlled by the user via the number of photons injected into the telescope by the ray-tracing module. The number of photons is determined by a trade-off between run time and the desired accuracy of the response function.

RMF (SXI and SXS only)
SXS: The script *sxsmkrmf* calculates weighting factors based on the resolution grade distribution and the region used to the extract the spectrum , and runs the *sxsrmf* to create a single net RMF using these weights. *sxsmkrmf* requires as an input the same event file as well as the same list of grade and pixel used to extract the spectrum. By default, the SXS RMF input and output grids result in an RMF with 32768 channels of width 0.5eV.

SXI: *sxirmf* uses the WMAP, a coarse-binned image of the extraction region with the spatial counts distribution, to determine the weight distribution for the LSF from different positions on the detector. The WMAP is created by *xselect* and stored in the primary header of the spectral file. By default, an RMF is generated with an input energy grid in the range 0.200 to 23.974 keV, and with 5900 output channels of width 2 eV up to ~12 keV and 500 channels up to ~24 keV.

Exposure Map
The telescope effective area for a given fixed position on the SKY depends on the polar (off-axis) and azimuthal angle that the direction vector to that point makes with the telescope optical axis. The optical axis position varies in the SKY frame with time due to attitude variation. The effective area varies with the landing positions of photons on the focal plane, coming from the same fixed point on the SKY, and with time. *ahexpmap* organizes the attitude data into an histogram and for each of the histogram bin calculates GTI grouped by the off-axis and azimuthal angles of target X-ray source (or nominal pointing if there is no source), with respect to the telescope optical axis. For the SXS and SXI, the exposure map file also contains lists of pixels, for each attitude group (or histogram bin), that are not active for the full duration of that attitude group. Each pixel in the list is assigned a value (<1) that is equal to the fraction of time the pixel is active (or "ON") out of the total duration of the of the attitude group the pixel belongs to (a pixel can belong to any number of attitude groups, but will only be listed in the exposure map if its active fraction is <1).
Note that the number of discrete attitude bins should not be too large (generally <20), otherwise the run time for the ARF generator may become too large. If the number of attitude bins is too large, it should be check that the GTI intervals correspond to events that were cleaned with a sufficiently narrow range in the attitude fluctuations (i.e. exclude excursions that make the off-axis angle more than a few arcmin). In addition to, or instead of, curbing the attitude fluctuation amplitude, the tool *ahexpmap* could be run with a different setting of the parameters delta and/or numphi. If *ahexpmap* is run with 'delta=20' and 'numphi=1' there is, effectively, a single attitude bin in the output histogram. It is necessary to run *ahexpmap* for all imaging instruments (HXI SXI and SXS) before to create the RSP (HXI) or ARF (SXI and SXS).

CAMS delta-attitude file (HXI only)
The HXI detectors have residual motion relative to the telescope optical axis due to movement of the extended optical bench. These movements are tracked by the CAMS laser alignment system. The effective area calculation for the response matrix (RSP) must account for the fact that ray-tracing photons land on a pixel on the detector that changes with the EOB movements, using the CAMS information, in addition to the off-axis and azimuthal angles. The task *cam2att* calculates the necessary files to account for this motion and contain the position of the optical axis in RAWX and RAWY detector pixel coordinates, as a function of time necessary in the RSP

calculation.

ARF concept

Running aharfgen

The script *aharfgen* takes as input an exposure map file created by *ahexpmap*, and (optionally) a source selection region in SKY (SXI and HXI) or DET (SXS) coordinates. If a sky region is specified *aharfgen* makes several new region files in detector coordinates, one region file for each discrete attitude histogram bin in the exposure map file. *aharfgen* then sequentially runs the ray-tracing code *xrtraytrace* with parameters appropriate for each attitude bin, and stores the results in a single file. There are four options for the treatment of the spatial distribution of the X-ray source. The method is specified by the *aharfgen* parameter 'sourcetype'. One option is for a point source at infinity ('sourcetype = point'), the other three options are for extended sources. These are :

- 'sourcetype=flatcircle': Extended source at infinity that has a spatial distribution that has uniform flux over a circular region (zero outside of the circle). The radius of the source is specified by the value of the parameter 'flatradius', in arcmin.
- 'sourcetype=betamodel': Extended source at infinity that has a spatial distribution described by the beta model. Three numbers in the string input parameter 'betapars' specify the model. These are the core radius, the index (beta), and the maximum radius.
- sourcetype=image: A FITS image file (in RA/Dec coordinates) is used as input to the simulator *heasim*, to create a list of simulated events which are input to the ray-tracing for each of the attitude bin. The input image file may be from the SXI or from a different mission, or a model. If an SXI image is used, the results are not as accurate compared to an higher spatial resolution image (a *Chandra* image file for example), because photons in the SXI have already passed through the X-ray telescope, which has a PSF half-power diameter of ~1.3 arcmin. The name of the image file is given by the parameter *imgfile*. There is an implicit assumption that the spatial flux distribution in the input image file is identical for all energies over the range that the effective area and response function is being calculated.

For extended source (e.g. sourcetype set to FLATCIRCLE or BETAMODEL or IMAGE), the effective area in the ARF (or RSP) file is different to that of a point source because for extended sources, incident photons enter the telescope with a range of off-axis angles, resulting in different efficiencies for impacting the focal-plane at a given energy. When used in spectral fitting, the ARF or RSP for an extended source gives a flux that corresponds to the full spatial extent of the source. For sourcetype=FLATCIRCLE or sourcetype=BETAMODEL, this corresponds to the model flux out to the maximum radius parameter ("flatradius" or the third number in "betapars" respectively), and for sourcetype=IMAGE it corresponds to the flux of the entire input image.

However, note that the effective area is normalized to the flux of the input image (not its size) and therefore no correction for the size of the image is necessary. So for example an input image of a single point source for sourcetype=IMAGE gives the correct flux for a point source regardless of the image size. For extended sources, if the selection region is smaller than the full source, it would be recommended to use an input image that includes 2-3 arcmin larger than the selected region. No vignetting correction should be applied to the flux obtained using an ARF or RSP for extended sources because it is by definition already in the effective area calculation.

Regardless of whether the ARF or RSP is for a point source or an extended source, the effective area already accounts for the finite size of the detector and any region selection used for spectral extraction.

The value of the *aharfgen* parameter *numphoton* is chosen considering the trade-off between statistical accuracy and the run time. The ray-tracing module is the limiting factor for the run time, which could become impractical if *numphoton* is too large. The input exposure map contains an attitude histogram and *numphoton* corresponds roughly to the number of ray-tracing photons allocated to each attitude bin per internal energy grid point. If any attitude bin has a time interval significantly larger than the average time interval, *aharfgen* allocates additional ray-tracing photons to that attitude position. The user does not have control over the energy grid points used for the ray-tracing model, aside from the minimum and maximum energy. The recommended strategy is to choose *numphoton* to give a run time in the 30-40 minute range, examine the statistical quality of the resulting ARF, and if necessary, re-run *aharfgen* with a larger value of *numphoton* (overnight if a substantially better statistical quality is desired). As a rule of thumb, the initial value of *numphoton* for the SXS and SXI is ~5000 (N_attitude/20)*[(energy range in keV)/16], where N_attitude is the number of attitude bins created by *ahexpmap*. For the HXI the initial value of *numphoton* for the HXI is then ~600 (N_attitude/20) x [(energy range in keV)/65], where N_attitude is the number of attitude bins created by *ahexpmap*.

The results of the ray-tracing are saved to an event file on disk. The photon paths, or events, along with the source and region information, are passed onto *ahsxtarfgen* (SXI or SXS) or *hxirspeffimg* (HXI) by *aharfgen* in order to enable the telescope effective area to be calculated. Detector efficiences are calculated in *ahsxtarfgen* and *hxirspeffimg*, not in *aharfgen*. The ray-tracing file that is saved to a local disk may be re-used. Note that when *aharfgen* is run, and it finds that a ray-tracing file with the requested file name already exists, *aharfgen* skip running the ray-tracing code again, and instead pass on the existing event file onto *ahsxtarfgen* or *hxirspeffimg*, so that unnecessarily long run times may be avoided.

Energy Range and Energy Grid

aharfgen creates an ARF (or RSP for HXI) that has an energy grid identical to the input RMF. In the case of the SXS or SXI, this input RMF is user supplied. *aharfgen* only uses the SXS or SXI RMF to read the energy grid, and does not use the matrix data at all, so the input RMF does not need to be made from the event file specific to the observation and can be very generic. This is not the case for the HXI, where *aharfgen* uses the energy grid in the CALDB LSF file, so the user does not need to supply an additional RMF.

The valid energy ranges for which *aharfgen* can produce results wider than the ranges for which the instruments were reliably calibrated in pre-flight ground experiments. The valid and best-calibrated energy ranges are given for each instrument in the table below (Table 4.1).

aharfgen allows the user to specify the energy range over which to calculate the output ARF or RSP. The energy range has to be selected within the range of energies over which the telescope reflectivity, stored in CALDB, is calculated. Narrower energy ranges shorten the run time of the program, therefore it is worth considering to do so if the user project does not need the full energy range. However no matter what energy range is chosen, the output ARF or RSP is always be made on the full energy grid of the input RMF, but the effective area outside of the specified energy range is set to zero.

Instrument	*Valid Energy Range (keV)*	*Calibrated Energy Range (keV)*
SXS	*0.03 to 30*	*0.12 to 17*
SXI	*0.03 to 24*	*0.12 to 17*
HXI1, HXI2	*2 to 100*	*4 to 70*

Table 4.1: Valid and best calibrated energy ranges for each instrument on-board Hitomi

The ray-tracing energy grid is coarser than the final output energy grid because the ARF/RSP tools utilize some pre-calculated ray-tracing results to generate final effective areas on an arbitrarily fine energy grid. This way, a spectral resolution can be achieved that is more than sufficient for the SXS, yet avoiding impractical run times.

SGD RSP

The on-axis response file for each of CC of an SGD is provided in CALDB and if the source is off-axis, the task *sdgarfgen* computes an off-axis transmission and correct the instrument response. The transmission depends on source position because the SGD collimator partially blocks the light from off-axis sources. The computation is done by interpolating between points in FOC coordinates read from a CALDB file. The source position is be specified with the parameters 'RA' and 'Dec', and the satellite pointing is read from the input event file. The calculation does not consider the effect of pointing fluctuations because the size of these fluctuations (~few arcmin) is smaller than the typical scale of changes in the SGD collimator transmission ratio. *sgdarfgen* can calculate either a response for a single SGD CC or three separate responses for three CCs within an SGD.

Background

The background in all Hitomi detectors includes contributions from astrophysical sources, from high-energy particles interacting with the detector and spacecraft structure, and possibly from internal sources such as detector electronics or calibration sources illuminating the detector. The astrophysical background is due to X-rays that are focused by the telescope optics, and hereafter is called the X-ray background or XRB. The non-focused background components from particles and internal detector effects are collectively called the non-X-ray background or NXB. Because the spatial dependence of the XRB and NXB differ, some care must be taken to properly account for them. There are essentially the following two methods to treat the background, examples of which are detailed in Section 9:

(1) Subtract a local background spectrum. This spectrum can be extracted from an annulus around a point-like source or from an off-source region (for the SXI or HXI), or from an offset pointing (for SXS, since the field of view is small compared to the telescope PSF; or for an extended source in SXI or HXI). Since it contains both the contributions of the XRB and NXB, one simply needs to subtract it from the source spectrum during spectral fitting, e.g., with the 'back background.pha' command in XSPEC. The benefit of this technique is that spectral fitting is quite simple, with only a single source spectrum and model. The drawback of the technique is that is does not account for spatial variations of the background components. In particular, a

background extracted from a different region of the detector than the source will have a different XRB component due to telescope vignetting, and possibly a different NXB, and the components will not necessarily differ in the same way. If the background region is reasonably close to the source region, or extracted from the same detector region of an offset pointing in the same part of the sky, then this technique is probably safe to use.

(2) Subtract the NXB and simultaneously fit the source and XRB spectra. For this method, the user should again extract a background spectrum from a source-free region of the field of view or from an offset pointing. This spectrum is now treated as an additional source spectrum, where the source is the XRB. The user should then generate NXB spectra for each source and XRB spectrum using the tool for each instrument: *sxsnxbgen* (SXS), *sxinxbgen* (SXI), or *hxinxbgen* (HXI). The appropriate source or XRB region must be used as input to the NXB tool to account for any variations across the detector. In XSPEC, the corresponding NXB spectrum is subtracted from each source and XRB spectrum with the back background.pha' command. Then the source and XRB spectra are fit simultaneously with some background emission model, typically including: (a) an absorbed power-law (or power-law with a cutoff for the HXI) to account for unresolved point sources that make up the cosmic X-ray background (CXB); (b) an absorbed thermal component to account for the Galactic Halo (GH) and disk; (c) an unabsorbed thermal component to account for the Local Hot Bubble (LHB); and (d) additional thermal components or emission lines to account for other background sources like Solar Wind Charge Exchange (SWCX). More details can be found at the NASA HEASARC X-ray background help page (`http://heasarc.gsfc.nasa.gov/Tools/xraybg_help.html`). Note that since the XRB region is considered another X-ray source in this case, response files need to be generated for it as well. The benefit of this technique is that it most accurately accounts for both the XRB and NXB, and it can be used with background regions in very different locations on the detector compared to the source. The drawback is that spectral modeling is more complicated, with additional model components that need to be tied between spectra, and additional RMF and ARF that need to be generated.

4.4.3 Timing Analysis

Light curve may be extracted within *'xselect'* similarly to the spectrum. The region selection described in the spectral analysis is also valid to extract light curve. *'xselect'* allows to extract binned light curves as well as events within the extracted region. Either an event file or a binned light curve may be input to the *xronos* routines to fold the data or calculate a power spectrum.

If the absolute timing is required by the science analysis, the light curve and/or the event file times may be corrected at the barycenter of the solar system using the task *'barycen'*. This is multi-mission task may be run on any file also for example housekeeping data that have a time column.

5 SOFT X-RAY SPECTROMETER (SXS) DATA ANALYSIS

5.1 Introduction

The SXS consists of 36 microcalorimeter detectors arranged in a 6 × 6 array of 0.5 arcmin pixels. One of the pixels (pixel 12) is inactive; in its place is a dedicated calibration pixel offset from the main array. The SXS system also includes one of the two Hitomi Soft X-ray Telescopes (SXT-S), and the multi-stage cooling system that maintains the detector at the 50 mK temperature necessary to realize the high SXS energy resolution (~4.7 eV). The SXS is unique among the Hitomi instruments in terms of its small (~3 × 3 arcmin) field-of-view and large pixels, as well as its energy resolution. SXS events are classified by the onboard processing into 5 resolution grades, determined by the combination of time interval to the nearest event (High, Mid, Low), and the time interval to the event that immediately follows (Primary, Secondary). As described below, the dependence of the detector response on resolution, grade, and pixel must be accounted for when calibrating and screening SXS events, extracting data products, and constructing spectral responses.

5.2 Cleaned event file content

The files included in the standard data download in the directory sxs/event_uf were processed through the pipeline and calibrated accordingly. The primary calibration steps are summarized in Table 5.1. These were applied to the first files that were converted from Hitomi telemetry, and had time assigned and various keywords set.

Calibration Step	Tool	Description
Calculate MXS GTI*	`mxsgti`	MXS on/off for indirect and direct modes
Assign antico PI	`sxsanticopi`	Only antico calibration step
Assign coordinates	`coordevt`	Up to, and including, sky coordinates
Flag pixels*	`sxsflagpix`	Flag SXS events for antico, and MXS event coincidence, temporal proximity, crosstalk, lost gti
Associate SXS secondary events with primary	`sxssecid`	Identifies groups of events needed by sxsseccor
Compute the energy scale correction*	`sxsgain`	Run for the Cal-pixel MnKα feature, and each MXS feature
Assign PI	`sxspha2pi`	Apply pixel 12 energy scale correction to all events
Flag pixels*	`sxsflagpix`	Flag SXS events for antico, and MXS event coincidence, temporal proximity, crosstalk, lost gti
Associate SXS secondary events with primary	`sxssecid`	Identifies groups of events needed by sxsseccor
Apply secondary correction	`sxsseccor`	Adjust PHA for MS events
Assign PI	`sxspha2pi`	Apply pixel 12 energy scale correction to all events, with additional scaling and secondary corrections
Correct PI	`sxsperseus`	Apply Perseus energy scale correction to events,;applicable only to Perseus data

Table 5.1: Primary calibration steps. Note that the pipeline runs sxsflagpix, sxssecid, sxsseccor, and sxpha2pi twice since sxsflagpix requires PI to flag crosstalk from recoil with Cal-pixel events, and there are options in sxssecid and sxsgain to check this flag. *Since the MXS was not used, all or part of these tasks are no longer relevant.

The SXS event file columns relevant for screening, selection, and event extraction are listed in Table 5.2. The ITYPE column in the event file provides a numerical representation of the event grade (Table 5.3): values of 0, 1, 2, 3, 4 correspond to HP (high-resolution primary), MP (mid-resolution primary), MS (mid-resolution secondary), LP (low-resolution primary), and LS (low-resolution secondary) event grade types, respectively. ITYPE=5 designates baseline (BL) events, which are used for diagnostic purposes. ITYPE=6 is used for lost (EL) events, time intervals when events could not be processed by the on-board software. ITYPE=7 (Rj) is used for rejected events.

Column	Description	Range
ITYPE	Resolution Grade; ITYPE=0: HP, ITYPE=1: MP, ITYPE=2: MS, ITYPE 3: LP, ITYPE 4: LS, ITYPE 5: BL, ITYPE 6: EL, ITYPE 7: Rj	0-7
PIXEL	Pixel number (pixel 12 is the calibration pixel)	0-35
RISE_TIME	Measured time from baseline to peak for SXS pulse	0-255
PI	Linearized Energy Channel	0-32767
STATUS	16 bit Event Flag (14 in use)	
	STATUS[1]: in (0) or out (1) of all-pixel GTI file	0-1
	STATUS[2]: in (0) or out (1) of individual-pixel GTI	0-1
	STATUS[3]: coincident with antico events (1)	0-1
	STATUS[4]: coincident with other event within a temporal proximity	0-1
	STATUS[5]: coincident with pixel 12 event	0-1
	STATUS[6]: coincident with pixel 12, and recoil energy test satisfied	0-1
	STATUS[7]: coincident with event in wiring proximity (electrical crosstalk) – short timescale	0-1
	STATUS[8]: largest PHA in electrical crosstalk group – short timescale	0-1
	STATUS[9]: coincident with MXS, direct mode*	0-1
	STATUS[10]: coincident with MXS afterglow, direct mode*	0-1
	STATUS[11]: coincidence with MXS, indirect mode*	0-1
	STATUS[12]: coincident with MXS afterglow, indirect mode*	0-1
	STATUS[13]: coincident with event in wiring proximity (electrical crosstalk) – long timescale	0-1
	STATUS[14]: largest PHA in electrical crosstalk group – long timescale	0-1

Table 5.2: SXS event file columns relevant for calibration, screening, and filtering. *Since the MXS was not used, these are no longer relevant.

	$t_{prev} <= \delta t_2$	$\delta t_2 < t_{prev} <= \delta t_1$	$\delta t_1 < t_{prev}$

$t_{next} <= \delta t_2$	LS	LS	LP
$\delta t_2 < t_{next} <= \delta t_1$	LS	MS	MP
$\delta t_1 < t_{next}$	LS	MS	HP

Table 5.3: Definition of SXS event grades based on times to the previous (t_{prev}) and next (t_{next}) events, where δt_1=69.92 ms and δt_2 = 17.52 ms.

The cleaned event files in the sxs/event_cl directory are derived from the unfiltered events files in sxs/event_uf using two broad classes of screening, event-by-event and by good-time intervals (GTI); see Table 5.4 below for details. ITYPE>4 events are excluded in the event-by-event screening, as are events with anomalous pulse shapes, and those flagged for antico coincidence, temporal proximity, recoil crosstalk, or falling within a lost event GTI for the event pixel. The GTI screening filters slew and bad attitude intervals, intervals of saturated telemetry, intervals of gain (temperature) instability during the ADR cycle, and intervals when any MXS is on. The GTI files needed for this screening are already present in the auxil directory or sxs/event_uf directories, and (except for the lost GTI) are applied to all pixels. In addition, GTI corresponding to nominal instrument status based on housekeeping data encapsulated in the makefilter (mkf) file in the auxil directory, and GTI based on orbit- and pointing-derived criteria (times of stable pointing, unblocked by the earth, outside of regions of high background) encapsulated in the extended housekeeping (ehk) file in the auxil directory are also applied. The current screening criterion is based on an approximate characterization of the SAA that may be improved in the future by utilizing the variation of the rate of detected antico events in some energy range. The GTI constructed from ehk and mkf files may be remade in the course of re-screening the data – either automatically (if using the *ahpipeline* script), or manually. Event, mkf GTI, and ehk GTI screening criteria are contained in the select file (Table 5.5). The mkf file also includes provisions for making GTI per pixel, and the pipeline runs the *sxsxpixgti* task to effect this, and also merges these GTI with the lost event GTI in each pixel. The output file, which has the form OBSIDsxs_px1010_exp.gti and includes an extension of "bad" GTI (inverted GTI) required by the exposure map generator, is not applied as part of the screening, but must be accounted for in creating the exposure map and ARF file (Section 5.6.2). That is, lost events (ITYPE==6) and events in per pixel lost GTI are screened out, but the resulting reduction in exposure per pixel is accounted for in post-processing.

Type	**File**	**Criterion**	**Comments**
event-based	event_uf/OBSIDsxs_uf.evt	`ITYPE<5 &&`	Indicates X-ray event
		`(SLOPE_DIFFER==b0||`	Indicates X-ray event
		`PI>25000) &&`	
		`QUICK_DOUBLE==b0 &&`	Indicates single X-ray event
		`STATUS[3]==b0 &&`	Not antico coincidence
		`STATUS[4]==b0 &&`	Not temporal proximity event
		`STATUS[6]==b0 &&`	Not recoil crosstalk
		`STATUS[2]==b0 &&`	Not in per-pixel lost GTI
		`PI>600 &&`	Eliminates cross-talk events
		`RISE_TIME<127 &&`	Indicates X-ray event
		`PIXEL!=12 &&`	Not the calibration pixel

		`TICK_SHIFT>-8&&TICK_SHIFT<7`	Indicates X-ray event
GTI-based	auxil/OBSID.mkf	`HK expression`[1]	Nominal instrument status

GTI-based	auxil/OBSID.ehk	`ANG_DIST<1.5 &&`
		`SAA_SXS==0 &&`
		`T_SAA_SXS>436 &&`
		`ELV>5 &&`
		`DYE_ELV>5`
			Pointing <1.5′ of mean
			Satellite outside SAA
			Time since SAA > 436s
			Pointing > 5° above Earth & >5° above sunlit limb

GTI-based	auxil/OBSID_gen.gti	inside GTIPOINT and GTIATT	Excludes slew and bad attitude intervals
GTI-based	auxil/OBSIDsxs_tel.gti	inside GTITEL	Excludes times of telemetry saturation
GTI-based	$LHEA_DATA/ahsxs_adr.gti	inside GTIADROFF	Excludes times of unstable gain during ADR cycle
GTI-based	event_uf/OBSIDsxs_mxfn.gti*	inside GTIMXSFNOFF13 and GTIMXSFNOFF24	Outside intervals of MXS events, direct (MXS-1 and -3) and indirect (MXS-2 and -4)
GTI-based	event_uf/OBSIDsxs_el.gti	inside GTILOSTOFF	Outside lost event GTI; applied to antico events only [screening described above applied to pixel events only]

Table 5.4: Default event-based and GTI-based filtering criteria. *Since the MXS was not used, these are no longer relevant.

The processing and screening summarized in this section results in an event file suitable for extraction of data products. However, reprocessing of SXS events ought to be applied first if: (1) there are changes in any of the relevant CALDB files used in processing and screening (see explanation of CALDBVER in Section 3.2), (2) screening criteria different from the standard are applied, or (3) PI is reassigned. The user should ascertain whether any SXS calibration files relevant to the time of the observation have changed since the observation was processed by checking http://heasarc.gsfc.nasa.gov/docs/hitomi/calib/ and the list of calibration files used in the SXS pipeline processing shown in Table 5.5.

5.3 Reprocessing events

[1] Expression is:

```
HKSXSAA001==0&&HKSXSAB001>0&&((HKSXSAB002==1)&&(HKSXSAB003==1))&&((0.048<HKSX
SAC001&&HKSXSAC001<0.052)||(0.048<HKSXSAC002&&HKSXSAC002<0.052))&&((0.048<HKS
XSAC003&&HKSXSAC003<0.052)||(0.048<HKSXSAC004&&HKSXSAC004<0.052))&&HKSXSAD001
>0&&HKSXSAE001>0&&(129.0<HKSXSAD002&&HKSXSAD002<=131.0)&&(129.0<HKSXSAE002&&H
KSXSAE002<=131.0)&&(HKSXSAD003==1&&HKSXSAD004==1)&&(HKSXSAE003==1&&HKSXSAE004
==1)&&(HKSXSAF001==1||HKSXSAF002==1)&&((HKSXSAF003==1||HKSXSAF004==1))&&((HKS
XSAF007==1||HKSXSAF008==1))&&((HKSXSAI002==3||HKSXSAJ002==3))&&((HKSXSAM002==
3||HKSXSAN002==3))&&((HKSXSAI003==3||HKSXSAJ003==3))&&((HKSXSAM003==3||HKSXSA
N003==3))
```

(1) Reprocessing of the unfiltered SXS events file is necessary if updates to relevant CALDB files were made since the cleaned files were produced. The calibration files used in SXS pipeline processing are listed in Table 5.5.

Tool	Calibration File	Description
mxsgti*	ah_gen_delay_yyyymmddvNNN.fits	Instrument time delay file
coordevt	ah_sxs_teldef_yyyymmddvNNN.fits	SXS coordinates, alignment
sxsanticopi	ah_sxs_gainant_yyyymmddvNNN.fits	Antico gain
sxsflagpix	ah_sxs_coeftime_yyyymmddvNNN.fits ah_sxs_pixmap_yyyymmddvNNN.fits	Time intervals defining coincidence Pixel Numbering and wiring proximity
sxssecid	ah_sxs_coeftime_yyyymmddvNNN.fits	Time intervals defining grades
sxsseccor	ah_sxs_secpulse_yyyymmddvNNN.fits	MS pulse amplitudes and offsets
sxsgain	ah_sxs_gainpix_yyyymmddvNNN.fits ah_gen_linefit_yyyymmddvNNN.fits	SXS gain coefficients per pixel and grade Theoretical calibration line profiles
sxspha2pi	ah_sxs_gainpix_yyyymmddvNNN.fits	SXS gain coefficients per pixel and grade
ahfilter	ah_gen_mkfconf_yyyymmddvNNN.fits	Makefilter file for screening
ahgtigen	ah_gen_select_yyyymmddvNNN.fits	Selection file for screening

Table 5.5: SXS CALDB files used in calibration and screening. *Since the MXS was not used, these are no longer relevant.

If the *ahpipeline* script is run, the event files for all of the Hitomi instruments are reprocessed, and there is no need to conduct an SXS-specific reprocessing. To fully reprocess only the SXS event files, run the *ahpipeline* script with ‘instrume=SXS’ instead of ‘instrume=ALL’ as follows. From the directory above where the original data is located, and for an observation with OBSID ah100050020, issuing the command

```
%>ahpipeline indir=100050020 outdir=100050020_repro \
steminputs=ah100050020 stemoutputs=DEFAULT entry_stage=1 \
exit_stage=2 instrument=ALL verify_input=no create_ehkmkf=yes
```

applies all of the calibration steps summarized in Table 5.1 to reconstruct the unfiltered event file, and then all of the screening steps summarized in Table 5.4 to reconstruct the cleaned event files. These files are placed in the “repro” subdirectory indicated by the ‘outdir’ parameter. If the ‘entry_stage’ parameter is 2, the data is cleaned but not recalibrated; if the ‘exit_stage’ parameter is 1 the data is recalibrated but not cleaned. If the exit_stage parameter is 3 the pipeline data products are remade.

Alternatively, if the ehk and mkf files do not need to be remade, run the *sxspipeline* script as follows. From the directory above where the original data is located, and for an observation with OBSID ah100050020, issue the command

```
%>sxspipeline indir=100050020 outdir=100050020_repro1_sxs \
steminputs=ah100050020 stemoutputs=ah100050020 entry_stage=1 \
exit_stage=2 attitude=100050020/auxil/ah100050020.att.gz \
orbit=100050020/auxil/ah100050020.orb.gz \
obsgti=100050020/auxil/ah100050020_gen.gti.gz \
housekeeping=100050020/sxs/hk/ah100050020sxs_a0.hk1.gz \
makefilter=100050020/auxil/ah100050020.mkf.gz \
extended_housekeeping=100050020/auxil/ah100050020.ehk.gz
```

to recalibrate and rescreen the SXS data.

In cases of relatively low source count rates, most lost events occur during SAA passages. As a result there is little or no pixel-to-pixel variation in the lost GTI intervals (this may be checked by examining the event_uf/OBSIDsxs_px1010.gti file); moreover these intervals are excluded using other criteria (i.e., the ehk SAA screening described above). In this case one may account for the lost event GTI at the screening stage as follows:

```
%>sxspipeline indir=100050020 outdir=100050020_repro2_sxs \
steminputs=ah100050020 stemoutputs=ah100050020 entry_stage=1 \
exit_stage=2 attitude=100050020/auxil/ah100050020.att.gz \
orbit=100050020/auxil/ah100050020.orb.gz \
obsgti=100050020/auxil/ah100050020_gen.gti.gz \
housekeeping=100050020/sxs/hk/ah100050020sxs_a0.hk1.gz \
timfile=100050020/auxil/ah100050020.tim.gz \
extended_housekeeping=100050020/auxil/ah100050020.ehk.gz \
makefilter=100050020/auxil/ah100050020.mkf.gz screenlost=yes
```

By setting the ‘screenlost’ parameter to ‘yes’ the lost event GTI includes intervals of lost events for *any* pixel, and its application in the pipeline excludes these for *all* pixels. In this case, the pixel GTI file should not be used in the generation of the exposure map.

An additional task, *sxsperseus*, is run on SXS event files for the Perseus cluster observation to correctly assign PI; *sxspipeline* and *ahpipeline* automatically include this step and reprocessing can proceed in the usual way. The very first Perseus sequence is a special case that requires setting non-default *sxsgain* parameter as follows:

```
ahpipeline indir=100040010 outdir=100040010_repro3_sxs \
steminputs=ah100040010 stemoutputs=DEFAULT entry_stage=1 \
exit_stage=2 instrum=SXS verify_input=no create_ehkmkf=yes \
extraspread=100
```

(2) The methodology for applying different screening criteria than the standard screening depends on the nature of the difference. A stricter cut on columns in the extended housekeeping, makefilter, or event file columns can be applied to the cleaned data using *xselect*. An example is shown below for the case of filtering time of high cutoff rigidity. A more general rescreening requires a multi-step reprocessing of the unfiltered event files (see, also, Appendix A), as in the following example.

For the SXS, an alternative label (PIXELALL, as opposed to the default PIXELALL label) is included in the makefilter file that applies an additional screening based on measured temperature fluctuations (HKSXSAC010<2.5)&&(HKSXSAC011<2.5), and retains the cal-pixel events. The commands, to be issued from the 100050020/sxs subdirectory, needed to create such new cleaned event files is as follows:

a) Create the mkf GTI corresponding to the PIXELALL label

```
%>ahgtigen infile=../auxil/ah100050020.mkf.gz \
outfile=ah100050020_sxs_mkf.gti gtifile=NONE gtiexpr=NONE \
mergegti=AND selectfile=CALDB label=PIXELALL instrume=SXS prefr=0.0 \
postfr=1.0
```

b) Create the ehk GTI

```
%>ahgtigen infile=../auxil/ah100050020.ehk.gz \
outfile=ah100050020_sxs_ehk.gti gtifile=NONE gtiexpr=NONE \
mergegti=AND selectfile=CALDB label=PIXELALL instrume=SXS prefr=0.0 \
postfr=1.0
```

c) Create the text file sxs_ah100050020_gti.lst, which lists all of the SXS GTI file extensions to use in screening. The list includes the two made in the previous two steps, as well as the GTI in the original unfiltered event file, the non-saturated telemetry GTI, the pointing GTI, the good attitude GTI, the GTI when all MXS are off, and the nominal ADR GTI.

```
event_uf/ah100050020sxs_p0px1010_uf.evt.gz+2
../auxil/ah100050020sxs_tel.gti.gz+1
../auxil/ah100050020_gen.gti.gz+2
../auxil/ah100050020_gen.gti.gz+5
event_uf/ah100050020sxs_mxfn.gti.gz+11
event_uf/ah100050020sxs_mxfn.gti+12
ah100050020_sxs_ehk.gti+1
ah100050020_sxs_mkf.gti+1
event_uf/ahsxs_adr.gti+2
```

d) Create a GTI file that merges all the GTI in the list in step (c):

```
%>ahgtigen infile=NONE outfile=ah100050020sxs_screen.gti \
gtifile=@sxs_ah100050020_gti.lst gtiexpr=NONE mergegti=AND
```

e) Screen the data using the merged GTI file created in step (d), as well as (for example) the updated event screening described above:

```
%> ahscreen infile=event_uf/ah100050020sxs_p0px1010_uf.evt.gz \
outfile=ah100050020sxs_p0px1010_cl2b.evt \
gtifile=ah100050020sxs_screen.gti \
expr="ITYPE<5&&(SLOPE_DIFFER==b0||PI>25000)&&QUICK_DOUBLE==b0&& \
STATUS[3]==b0&&STATUS[4]==b0&&STATUS[6]==b0&&STATUS[2]==b0&&PI>600 \
&&RISE_TIME>=40&&RISE_TIME<127&&PIXEL!=12 \
&&TICK_SHIFT>-8&&TICK_SHIFT<7" mergegti=AND selectfile=CALDB \
label=NONE cpkeyword=all \
```

f) Adjust the PI TLMIN and TLMAX keywords (the unfiltered event file included baseline, ITYPE=5, events with PI that may be <0).

```
%>fthedit ah100050020sxs_p0px1010_cl2b.evt TLMIN45 a 0
%>fthedit ah100050020sxs_p0px1010_cl2b.evt TLMAX45 a 32767
```

Note that one could add additional GTI at step a, b, or c; remove some of the standard GTI screening at step d, or add additional expressions at steps a, b, or d.

(3) Reprocessing is necessary in order to reassign PI using a different method, or different parameters, than the standard pipeline processing. PI in the original unfiltered and cleaned events files is assigned based on application of the gain history file (ghf) calculated for the cal-pixel (pixel 12) to events in all pixels, with additional correction for MS events and a.[1] The ghf calculated for the MXS can be used to assign PI on a pixel-by-pixel basis, and is expected to be more accurate. The data may be recalibrated (and rescreened) using the MXS method to assign PI by explicitly setting the 'linetocorrect' parameter to, e.g. CuKa – the strongest of the MXS features – in *sxspipeline* (by default 'linetocorrect=MnKa', and the cal-pixel method is used) as follows:

```
%>sxspipeline indir=100050020 outdir=100050020_repro3_sxs \
steminputs=ah100050020 stemoutputs=ah100050020 entry_stage=1 \
exit_stage=2 attitude=100050020/auxil/ah100050020.att.gz \
orbit=100050020/auxil/ah100050020.orb.gz \
obsgti=100050020/auxil/ah100050020_gen.gti.gz \
housekeeping=100050020/sxs/hk/ah100050020sxs_a0.hk1.gz \
timfile=100050020/auxil/ah100050020.tim.gz \
extended_housekeeping=100050020/auxil/ah100050020.ehk.gz \
makefilter=100050020/auxil/ah100050020.mkf.gz \
linetocorrect=CuKa
```

The gain files, if created, that are included in the event_uf directory are defined in Table 5.6 for OBSID ah1000506287; however, note that the Mg Kα and Al Kα lines from the indirect MXS operation should not be used for energy scale correction of a single observation.

Gain file name	Line	Method/purpose
ah000506287sxs_010_px12.ghf	MnKα	Calibration pixel based energy scale correction
ah000506287sxs_010_cua.ghf	CuKα	MXS (direct) pixel-by-pixel energy scale correction
ah000506287sxs_010_cra.ghf	CrKα	MXS (direct) pixel-by-pixel energy scale correction

[1] The remainder of this section is no longer relevant for HITOMI data since the MXS was not operated; however, note that if PI needs to be re-assigned for any reason (e.g., CALDB or software update*) sxspipeline* (and not sxspha2pi alone) should be used since the pipeline runs s*xspha2p*i twice.

ah000506287sxs_010_mga.ghf	MgKα	MXS (indirect) pixel-by-pixel gain monitoring
ah000506287sxs_010_ala.ghf	AlKα	MXS (indirect) pixel-by-pixel gain monitoring

Table 5.6: Gain history file examples

5.4 Extracting products

This section summarizes the steps needed for the extraction of data products – images, spectra, and light curves – including additional screening, region and grade selection, and caveats to watch out for. See Section 9 for a summary of the parallel analysis steps for all Hitomi instruments.

5.4.1 Additional screening of events

1) Screening out electrical crosstalk events

The standard screening steps outlined in Section 5.2 do not account for electrical crosstalk events, which are false triggers induced by events in adjacent pixels as defined by the pixel wiring. The effect is mitigated with a cut on the pulse risetime applied to all events except those of grade LS, implemented by issuing the following command in the sxs/event_cl directory:

```
%>ftselect ah100050020sxs_p0px0000_cl.evt \
ah100050020sxs_p0px0000_cl2.evt \
"(RISE_TIME>=40 && RISETIME<=60 && ITYPE<4 && PI>=400) || (ITYPE==4 &&
PI>=400)"
```

This creates the "cleaned-2" event file ah100050020sxs_p0px0000_cl2.evt. The PI cut is not needed if that latest screening, which eliminates events with PI<600, was applied.

2) Filtering times of high COR

To reduce the non-X-ray background (NXB), a time filter can be applied to eliminate any data associated with high COR (Cut-Off Rigidity). The exact COR cut-off depends on the particular observation and science goals, however a rule of thumb is that low surface brightness observations may benefit from a stricter minimum COR cutoff, while for bright point sources such a restriction is less important, and it reduces the effective exposure time. The COR information is contained in the EHK file in a number of columns; the recommended value to use is COR2 as in the following pair of *xselect* commands that eliminates time intervals associated with locations in the orbit where COR2>6. :

```
xsel:HITOMI-SXS-PX_NORMAL>read hk hkfiles=../auxil/ah100050020.ehk
expand=yes
xsel:HITOMI-SXS-PX_NORMAL> select hk "COR2.GE.6"
```

The resulting filtered event file can be saved using `extract events/save events` within *xselect*.

3) Grade selection and caveats for high count rate sources

Generally, analysis with a significant spectroscopic component should be done with event files that are additionally filtered on grade (ITYPE) to include, at most, only HP, MP, and MS events. HP events have the best calibrated PI and sharpest energy resolution (see Table 5.7). The *sxsbranch* tool computes the distribution of grades over the SXS array, and provides a means to inform subsequent filtering. These fractions per grade – the branching ratio -- are recorded in the 'BRANCHHP', 'BRANCHMP', 'BRANCHMS', 'BRANCHLP', and 'BRANCHLS' keywords in the header of second extension ('BRANCHCALC') of the *sxsbranch* output file. For example, the command

```
%>sxsbranch ah100050020sxs_p0px0000_cl2.evt real \
sxsbranch.out pixfrac=$LHEA_DATA/pixfrac.txt pixmask=none \
calpixrate=6.0 ctphafrac1=0.0 ctphafrac2=0.0
```

stores these ratios, as well as theoretical estimates (in the 'BRANCHEST' extension) in the output file 'sxsbranch.out'. The estimated ratios are calculated ignoring the effects of crosstalk, assuming a cal-pixel count rate of 6 ct s^{-1} in the estimates, and a point-source-like distribution of flux ('pixfrac=$LHEA_DATA/pixfrac.txt)', and may be retrieved via the command

```
%>fkeyprint "sxsbranch.out[BRANCHCALC]" BRANCHHP
```

The estimated branching ratios, assuming no crosstalk, for a point source distribution centered on the array is shown in Figure 5.1 (left); the fraction of HP/MP events becomes significantly less than unity at SXS count rates of ~2 ct s^{-1} per pixel. The HP+MP ratio (on a log scale) with and without electrical crosstalk is also shown in Figure 5.1 (right). The latter is calculated assuming that <u>every</u> real event triggers a crosstalk event in pixels that are adjacent based on the pixel wiring. Also shown is the expected fraction of events that have another event within 2 ms. Since the SXS processor cannot trigger on additional pulses arriving within this time window following an event, this leads to a type of event pileup.

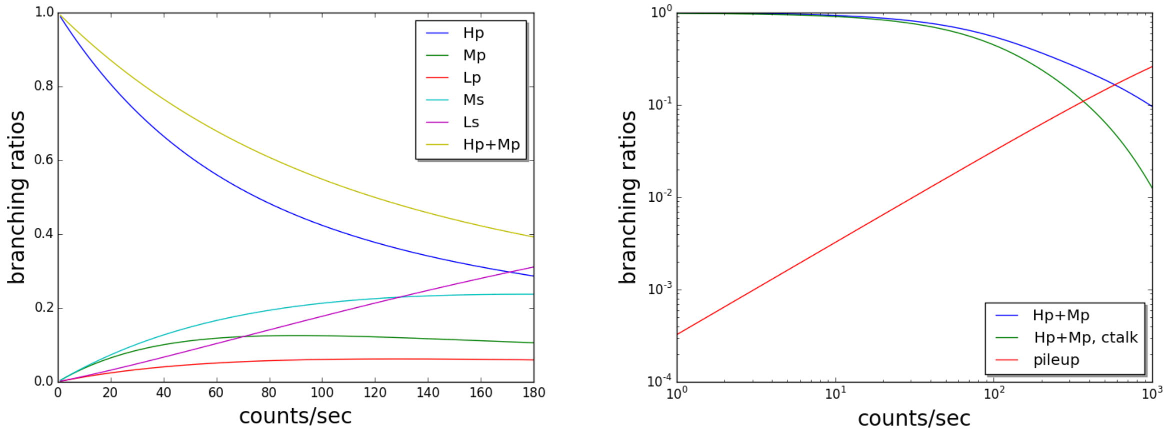


Figure 5.1: (Left) Branching ratios versus count rate in SXS array, neglecting crosstalk, estimated for a point source distribution. (Right) HP+MP branching ratio with (green curve) and without (blue curve) maximal electrical crosstalk between pixels adjacent according to their

wiring. Also shown (red curve, righthand plot) is the estimated fraction of events that have another event within 2 ms.

For very SXS high count rates (>200 ct s^{-1}), limitations in the number of events that can be processed by the SXS results in a decreasing livetime. The resulting distortion in branching ratios is reflected in the divergence of the values in the `BRANCHEST` and `BRANCHCALC` extensions.

The user may prefer to include only HP events if the highest spectral resolution is needed, or if the addition of MP and MS events does not significantly improve the statistics.

	GRADES	**XSELECT expression**
Pure timing/imaging	ALL	...
Spectroscopy	HP+MP+MS	filter GRADE 0,1 ,2
Highest resolution spectroscopy	HP	filter GRADE 0

Table 5.7: Grade filtering options

4) Pixel-based spatial filtering using sxsregext or xselect

The fundamental, irreducible spatial unit for the SXS is the physical pixel. A spectrum that is extracted from a region that geometrically only partially includes a pixel is ill-defined, and it is not possible to make an accurate ARF for such a spectrum. Therefore, SXS spectra and light curves should be extracted from a selection of full pixels, not fractions of pixels. The detector region used to extract a spectrum must match those used in deriving the ARF and RMF files (see below). For most purposes the entire SXS array, with the exception of the calibration pixel, should be used since the array is smaller than the extent of the point spread function. This choice optimizes the accuracy and reliability of the ARF.

The region used to extract spectra and light curves must be made of boxes that encompass individual pixels (see below). The *sxsregext* script enables the user to extract SXS products for an SXS event file using a sky or detector region. If the input region is in SKY (WCS ra and dec) coordinates, *sxsregext* proceeds by creating an SXS exposure map (see Section 5.6.2), deriving a list of pixels that have any portion within the SKY region for any of the bins in the exposure map pointing histogram, and constructing the corresponding region file in DET coordinates using the *ahmkregion* task. The input event file is then filtered on an input list of grades. An image extracted from the full array, and a lightcurve and spectrum extracted from the newly extracted region, are then produced. Examples are shown below.

The following section includes worked examples of the extraction of spectra and other data products using *xselect*. The sources considered are the compact quiescent source G21.5-0.9, a pulsar wind nebula observed by Hitomi, a simulated diffuse thermal source with turbulent broadening (cluster_sxs.fits), and a simulated bursting point source with peak to quiescent flux ratio of 30 (burst_sxs.fits). The branching ratios read from the headers of the output of running *sxsbranch* on these files is shown in Table 5.8.

	BRANCHHP	BRANCHMP	BRANCHMS	BRANCHLP	BRANCHLS
G21.5-0.9	0.984	0.005	0.006	0.0015	0.004
cluster	0.989	0.004	0.004	0.001	0.001
burst	0.296	0.076	0.154	0.040	0.433

Table 5.8: Fractions per grade from *sxsbranch*

5.4.2 Examples of extracting products

Images, light curves, and spectra are extracted from cleaned SXS event FTIS files of the observation, where any filtering not related to grade (ITYPE) and PIXEL (e.g., COR or RISE_TIME cut) is assumed to have been done in advance. If the observation is split into multiple event files, these may be combined in *xselect* prior to extraction. In the first example (G21.5-0.9), "cleaned-2" event files (Section 5.4.1) are constructed for data combined from three out of a total of five sequences.

The barycentric correction may be applied prior to extraction of data products using the *barycen* task, which requires an orbit file and source coordinates as input, as shown in the following example:

```
%>barycen infile=ah100050020sxs_p0px1010_cl.evt.gz
outfile=ah100050020sxs_p0px1010_cl_barycor.evt
orbfile=ah100050020.orb.gz ra=278.38824  dec=-10.570683 orbext=ORBIT
```

Note that the 'ra' and 'dec' parameter should be set to the true average pointing direction. This should be set to the RA_NOM and DEC_NOM keywords in the header for cases where the attitude is correct, or may be approximated by the object coordinates (RA_OBJ and DEC_OBJ header keywords) in cases where the source is known to be on-axis.

A) Quiescent Compact Source

Below are the *xselect* commands required for data product extraction for G21.5-0.9, a compact, quiescent source with an absorbed power-law or broken power-law spectrum, and 2-8 keV flux ~5.5 x10^{-11} erg cm^{-2} s^{-1}.

1. **Read in the SXI event list data, examine the 0.5-8 keV image**

```
xsel:SUZAKU > read events g21_sxi_cl.evt
xsel:HITOMI-SXI-WINDOW1 > set xybinsize 4
xsel:HITOMI-SXI-WINDOW1 > filter pha_cutoff 83 1333
xsel:HITOMI-SXI-WINDOW1 > extract image
xsel:HITOMI-SXI-WINDOW1 > plot image
```

The file g21_sxi_cl.evt is the cleaned SXI event file (see next section).

2. Visualization: Superimpose the SXS array

Compare the centroid of the source SXI image with the SXS array using `ahmkregion`.

```
xsel:HITOMI-SXI-WINDOW1 > $ahmkregion instrume=SXS ra=278.38824
dec=10.570683 roll=88.50244
```

where the ra and dec correspond to the centroid of the SXI image, and the roll angle is obtained from the event file PA_NOM header keyword, e.g.

```
fkeyprint g21_sxi_cl.evt PA_NOM
```

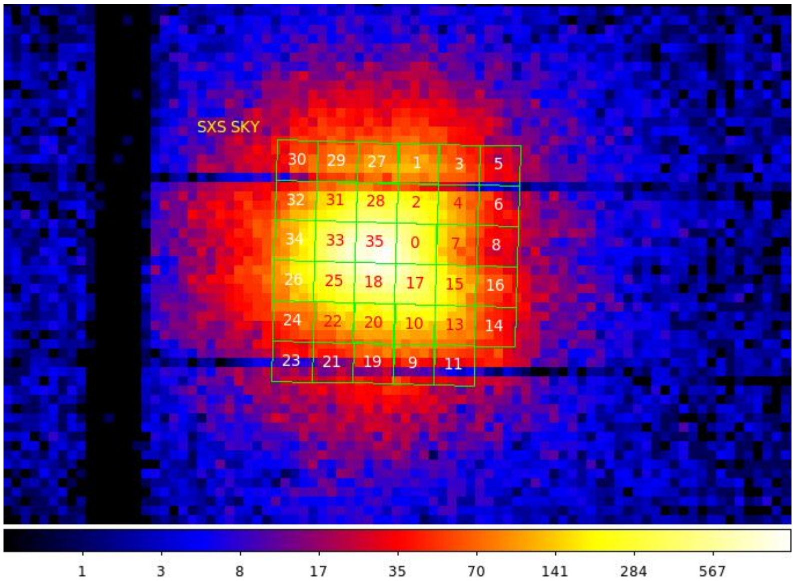


Figure 5.2: G21.5-0.9 SXI image with SXS pixels superimposed.

3. Extract the SXS data products, using the entire array

First, combine the cleaned event files and apply the RISE_TIME screening of Section 5.4.1.

```
xsel:SUZAKU > read events ah100050010sxs_p0px1010_cl.evt
xsel:HITOMI-SXS-PX_NORMAL > read events ah100050020sxs_p0px1010_cl.evt
xsel:HITOMI-SXS-PX_NORMAL > read events ah100050030sxs_p0px1010_cl.evt
xsel:HITOMI-SXS-PX_NORMAL > extract events
xsel:HITOMI-SXS-PX_NORMAL > save events
ah10005001230sxs_p0px1010_cl.evt
```

```
%> ftselect infile='ah10005001230sxs_p0px1010_cl.evt[events]' \
outfile=ah10005001230sxs_p0px1010_cl2.evt \
expression="(PI>=400)&&((RISE_TIME>=40 && RISE_TIME<=60 && \
ITYPE<4)||(ITYPE==4))"
```

The ehk files in the auxil subdirectory must also be combined.

```
%> ftmerge \
"ah100050010.ehk.gz,ah100050020.ehk.gz,ah100050030.ehk.gz"
ah10005001230.ehk
```

The *sxsregext* script is now run on the combined files.

```
%> sxsregext infile=ah10005001230sxs_p0px0000_cl2.evt \
regmode=SKY region=ah10005001230sxs_wcs.reg resolist=0 \
outroot=ah10005001230sxs_region_SXS_det \
outexp=ah10005001230sxs_expmap.fits \
ehkfile=ah10005001230.ehk delta=0.4 numphi=4
```

Here ah10005001230sxs_wcs.reg might be the identical extraction region used to extract the SXI light curve and spectrum and is assumed to include all 35 non-calibration pixels (as would any circular region centered on the SXS with radius greater than 1.5 arcmin). The output files from this command include the detector image, HP spectrum and light curve, exposure map, and detector region file created and used in the extraction (Figure 5.2) – in this case, ah10005001230sxs_region_SXS_det.pha, ah10005001230sxs_region_SXS_det.pha, ah10005001230sxs_region_SXS_det.pha, ah10005001230sxs_expmap.fits, and ah10005001230sxs_region_SXS_det .reg (see Figure 5.4), respectively. Note that, by default, HP (ITYPE==0) and MP (ITYPE==1) events are included in the spectrum; this is controlled via the hidden *sxsregext* 'resolist' parameter.

The same detector region and grade selection must be used as input to the SXS RMF and ARF generation scripts (see below). If the 'regmode' parameter is set to `DET`, the products are directly extracted using the DET region specified by the 'region' parameter.

The same spectrum and lightcurve may be extracted in *xselect* using the three individually "cleaned-2" event files as follows. Only HP events are used, as are all pixels except pixel 12 (these events are now removed as part of the standard screening).

```
xsel:SUZAKU > read events ah100050010sxs_p0px1010_cl2.evt
xsel:HITOMI-SXS-PX_NORMAL > read events
ah100050020sxs_p0px1010_cl2.evt
xsel:HITOMI-SXS-PX_NORMAL > read events
ah100050030sxs_p0px1010_cl2.evt
xsel:HITOMI-SXS-PX_NORMAL > filter column "PIXEL=0:11,13:35"
xsel:HITOMI-SXS-PX_NORMAL > filter GRADE "0:0"
xsel:HITOMI-SXS-PX_NORMAL > extract spectrum
xsel:HITOMI-SXS-PX_NORMAL > save spectrum
ah10005001230sxs_p0px1010_cl2_HP.pi
xsel:HITOMI-SXS-PX_NORMAL > extract curve
xsel:HITOMI-SXS-PX_NORMAL > save curve
ah10005001230sxs_p0px1010_cl2_HP.lc
```

The sole difference between 'ah100050020sxs_region_SXS_det.pha' created by *sxsregext* and ‘ah100050020sxs_p0px1010_cl2_HP.pi’ is that, in the former the BACKSCAL keyword is set to 5.468750E-01 (the ratio of the number of pixels used in the extraction, 35, to the total detector address space in pixels, 64) while in the latter BACKSCAL=1. This is not a problem provided that any spectra to be combined, subtracted, etc. are created in the same manner.

The resulting image and light curve are shown in Figure 5.3; the extracted spectrum is in Figure 5.4.

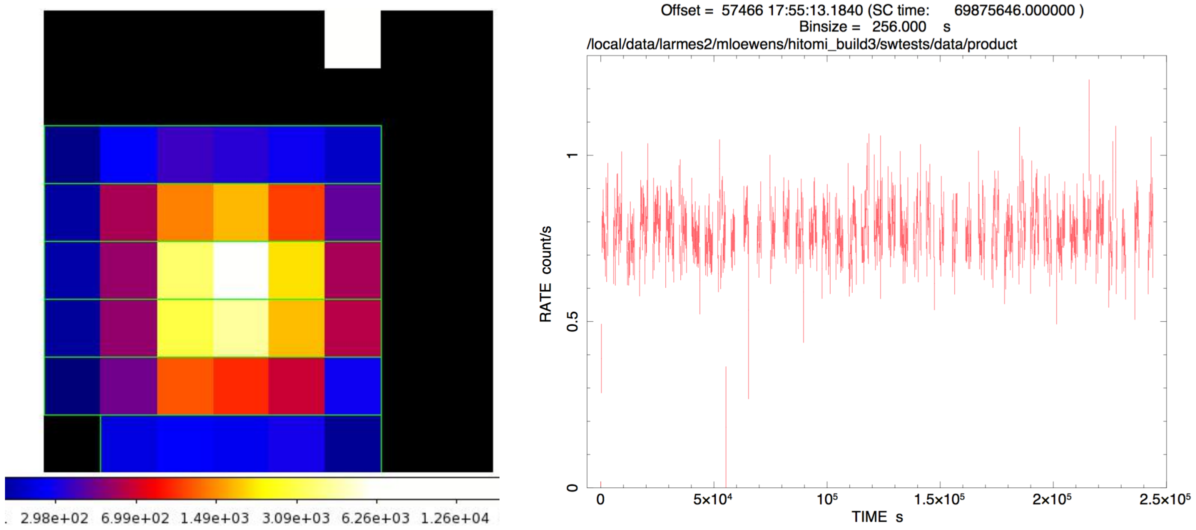


Figure 5.3: SXS G21.5-0.9 image in detector coordinates and extraction region created by sxeregext (6 rectangles outlined in green). In this case all pixels (except pixel 12) are included (left). G21.5-0.9 lightcurve from three combined sequences (right).

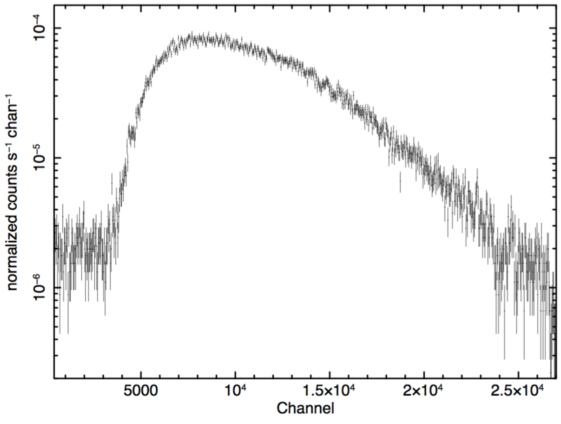


Figure 5.4: SXS G21.5-0.9 counts spectrum from three combined sequences .

B) Extended Source

This example uses a simulated 200 ks SXS observation of a cluster of galaxies with an isothermal plasma distributed following a beta-model surface brightness profile and typical temperature gradient with a cool core rising to a peak temperature and then falling off in the cluster outskirts. It is based loosely on results for Abell 1795, but with an additional velocity broadening of 200 km s^{-1}. This simulates the effects of turbulence as might be observed with the Hitomi SXS. The spectral extraction proceeds as in the previous example. Again, only HP events (ITYPE=0) are used – both because the extended nature of the source results in a high fraction of such events, and because this maximizes the sensitivity of the spectra to turbulent broadening. The steps are otherwise identical. The SXS array superimposed on the SXI image, and the SXS image, are shown in Figure 5.4. Only the high surface brightness, low-temperature core emission is within the SXS FoV. The entire spectrum, and spectrum in the region of the redshifted, broadened Fe K complex, are shown in Figure 5.5.

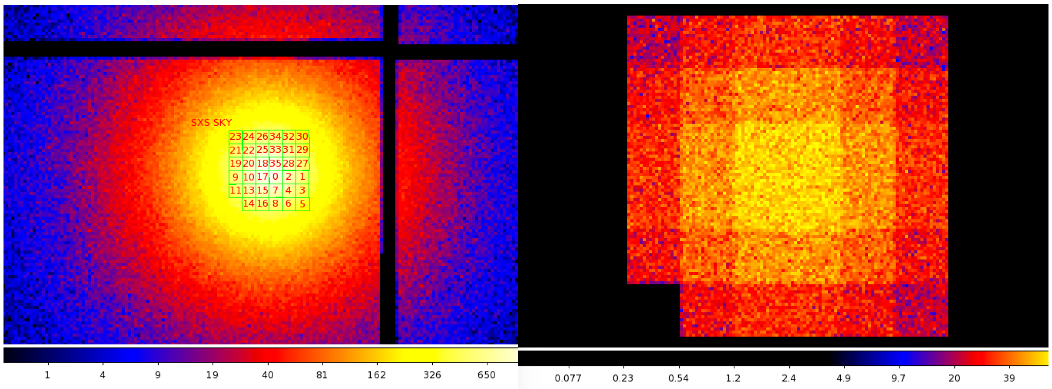


Figure 5.4: SXS array superimposed on the SXI image (left), and SXS image (right), for a simulated cluster.

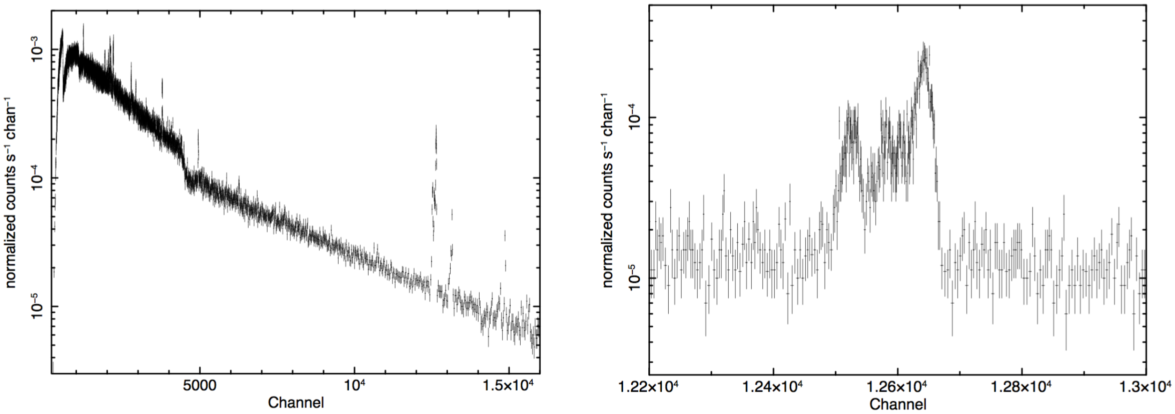


Figure 5.5: Simulated SXS spectrum: broadband (left) and Fe K complex region (right).

C) Burst Source

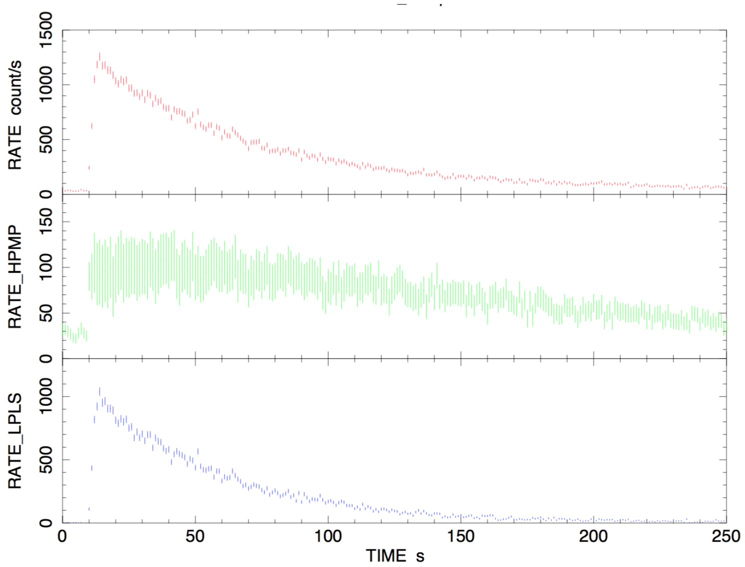
In this example light curves are extracted from a source with a burst initiating 10 s after the start of the 1 ks exposure, a linear risetime of 3 s, an exponential decay time of 1 minute, and a ratio of peak-to-quiescent flux of 30. Note that the branching ratios, which depend on the count rate, are time-dependent in this case. The ratios shown in Table 5.8 are *averages* over the entire 1 ksec exposure; the HP+MP fraction is lowest at the highest count rate. This may be illustrated by extracting light curves for different grade selection with consecutive runs of *sxsregext*, as follows:

```
%>sxsregext infile=burst_sxs.fits regmode=DET region=alldet.reg \
outroot=burst_allGrades resolist=ALL \
outexp=burst_sxs_expmap.fits ehkfile=burst_sxs.ehk delta=0.25 \
numphi=4

%>sxsregext infile=burst_sxs.fits regmode=DET region=alldet.reg \
outroot=burst_HPplusMP resolist=0,1 \
outexp=burst_sxs_expmap.fits  ehkfile=burst_sxs.ehk delta=0.25 \
numphi=4

%>sxsregext infile=burst_sxs.fits regmode=DET region=alldet.reg \
outroot=burst_LPplusLS resolist=3,4 \
outexp=burst_sxs_expmap.fits  ehkfile=burst_sxs.ehk delta=0.25 \
numphi=4
```

Figure 5.7 shows the sense in which the grade-filtered light curves are biased – with the burst diluted in the HP+MP curve (lower ratios at high count rates) and accentuated for the LP+LS curve.



.

Figure 5.2: Simulated SXS burst source light curves; top-to-bottom: total, HP+MP, LP+LS.

5.5 Background Estimation

Both external astrophysical and internal non-X-ray (NXB) components contribute to the total background associated with any observation. The local background, including the NXB, may be estimated using a spectrum extracted identically to the source region but from a separate off-source region. However due to the small SXS field-of-view, no such region is available unless a separate offset blank sky pointing is conducted. As a result, SXS spectral analysis should include subtracting the NXB background and simultaneously fitting the X-ray background and source spectra, i.e. modeling the X-ray background. The X-ray background should include a power-law representing the extragalactic background component, and one or more soft thermal components representing the Galactic and Local Hot Bubble background components.

The NXB spectrum may be estimated using the task *sxsnxbgen*. This task requires input source and NXB events files, input source and NXB EHK files, and either a region file or a list of pixels. The NXB files are located in the trend area. The NXB filtering must match that used to derive the source spectrum. Returning to the first example in Section 5.4, the command to create the NXB spectrum is:

```
%>sxsnxbgen infile=ah100050020sxs_p0px1010_cl.evt \
ehkfile=ah100050020.ehk  regfile=NONE  innxbfile=nxb.fits \
innxbehk=nxb.ehk outpifile=ah100050020sxs_p0px1010_nxb.pi \
sortcol="COR2" sortbin="0,4,5,6,7,8,9,10,11,12,13,99" \
pixels=“0-11,13-35” expr= “ITYPE==0 && RISE_TIME>=40 && RISETIME<=60”
```

The ‘sortcol’ and ‘sortbin’ parameters are chosen to match the COR2 filtering, the ‘pixels’ parameter the pixel selection, and the ‘expr’ parameter the additional filtering and grade selection, that were applied to the event file from which the source spectrum is extracted.

Note that, because of the short HITOMI lifetime, the sxsnxbgen tool is not relevant. Standard SXS background spectra based on observations with no source may be available in the future.

5.6 ARF and RMF generation

The following example illustrates the steps required to make an ARF and RMF for an SXS spectrum. A cleaned event file is required, as well as a list of pixels in the form of a region file in DET coordinates that defines the area from which the spectrum is extracted, the RMF generated, and the ARF defined.

5.6.1 RMF generation with *sxsmkrmf*

In order to analyze an SXS spectrum extracted and saved in *xselect*, one has to calculate a spectral response consistent with the pixel and grade selection that was applied in the spectral extraction. The RMF file is generated using the *sxsmkfrmf* script applied to the event file. The resulting RMF weighs contributions from each ITYPE specified in the *sxsmkrmf* `resolist` parameter, and each PIXEL to be included (see below), according to the counts in each (PIXEL, ITYPE) combination relative to the total counts in all selected pixels. *sxsmkfrmf* derives these weighting factors from the input event file, constructs a file containing them (`sxsfrac.fits’), and passes the file to the *sxsrmf* task which computes the RMF values. Here, the defaults for the input and output energy grids are used. This results in an RMF with an input energy grid in the range 0.0 to 16.384 keV, with 32768 channels, each of width 0.5 eV.

The input event file, and selection of pixels and grades input to the script, must be the same as was used to extract the spectrum. The pixel selection is applied by setting the *sxsmkfrmf* parameter ‘regmode=DET’ and the ‘region’ parameter to the output detector region file created by *sxsregext* (see above) as follows:

```
%> sxsmkrmf infile=ah10005001230sxs_p0px1010_cl2.evt \
outfile=ah10005001230_sxs_cl2_HP_medium.rmf resolist=0 regmode=det \
regionfile=ah10005001203sxs_region_SXS_det.reg whichrmf=m
```

The above command, by setting ‘whichrmf=m’, includes Gaussian core and exponential tail components. Since Hitomi operated with gatevalve closed and, thus, greatly reduced sensitivity below 2 keV, the above matrix is only applicable above about 2 keV. To extend below that, the electron loss continuum must be included by using the “x-large” option (which also includes the escape peaks). The threshold may be increased from 1.0e-9 to 1.0e-6 to reduce the size of the output file, with some loss of accuracy. The command is

```
%>sxsmkrmf infile=ah10005001230sxs_p0px1010_cl2.evt \
outfile=ah10005001230_sxs_cl2_HP_xlarge.rmf resolist=0 regmode=det \
regionfile=ah10005001230sxs_region_SXS_det.reg whichrmf=x \
rmfthresh=1.0e-6
```

5.6.2 ARF file generation using *ahexpmap* and *aharfgen*

The following example illustrates the steps to make an ARF for the SXS spectrum. The first step is to run *ahexpmap*, as in the following example. Some of the hidden parameters are also shown for completeness. The input parameters include the extended HK file for this observation (parameter ‘ehkfile’), the cleaned event file that provides the GTI (parameter ‘gtifile’), the GTI per pixel to be excluded (parameter ‘pixgtifile’), the attitude binning parameters ‘delta’ and ‘numphi’. Generally even if an exposure map was generated by running *sxsregext*, a new exposure map should be created to account for the GTI per pixel.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz  \
gtifile=ah100050020sxs_p0px1010_cl2.evt instrume=SXS badimgfile=NONE \
pixgtifile=ah100050020sxs_px1010_exp.gti.gz \
outfile=ah100050020sxs_p0px1010.expo outmaptype=EXPOSURE delta=0.25 \
numphi=4 stopsys=SKY instmap=CALDB qefile=CALDB contamifile=CALDB \
vigfile=CALDB obffile=CALDB fwfile=CALDB gvfile=CALDB maskcalsrc=yes \
fwtype=DEFAULT specmode=MONO specfile=spec.fits specform=FITS \
energy=1.5 evperchan=DEFAULT abund=1 cols=0 covfac=1
```

If the attitude is stable, or inaccurate based on a comparison of the centroid of the SXI image with the source coordinates, one can run *ahexpmap* with ‘numphi=1’ and a large value of ‘delta’ (~100) and so that there is a single attitude bin in the output histogram. Generally the ‘FRACTION’ column in the ‘OFFAXISHIST’ extension of the exposure map should be examined, and the value of ‘delta’ adjusted to eliminate entries with values <<1 to save computing time when constructing the ARF.

The next step is to run *aharfgen,* a script that internally calls *ahsxtarfgen*, to create the ARF file ah100050020sxs_p0px1010_point.arf. The following example produces an ARF for a point source (parameter 'sourcetype=POINT'). The input parameters include the source coordinates (parameters 'source_ra' and 'source_dec'), the exposure map produced in the previous step (parameter 'emapfile'), the extraction region in DET coordinates (parameter 'regionfile'), and the energy range (parameter 'erange'). If the attitude is suspect, the centroid of the source SXI image should be used in place of the true source coordinates. The expected runtime for the set of parameters below is ~30-40 minutes per attitude bin, and the value of 'numphoton' should be adjusted accordingly. An on-axis ARF may be generated by constructing an exposure may with a single attitude bin, and setting source_ra' and 'source_dec' to the values in the RANOMXP and DECNOMXP column.

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxs_ptsrc_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI instrume=SXS
emapfile=ah100050020sxs_p0px1010.expo regmode=DET \
regionfile=ah100050020sxs_region_SXS_det.reg sourcetype=POINT \
rmffile=ah100050020_sxs_cl2_medium.rmf erange="0.5 17.0 0 0" \
outfile=ah100050020sxs_p0px1010_point.arf numphoton=300000 \
minphoton=1 teldeffile=CALDB qefile=CALDB contamifile=CALDB \
obffile=CALDB fwfile=CALDB gatevalvefile=CALDB onaxisffile=CALDB \
onaxiscfile=CALDB mirrorfile=CALDB obstructfile=CALDB \
frontreffile=CALDB backreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB
```

The following example produces an ARF for a uniform circular distribution of 10 arcminute radius ('sourcetype=POINT', 'flatradius=10.0'), ah100050020sxs_p0px1010_flat.arf.

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxs_flat_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXS emapfile=ah100050020sxs_p0px1010.expo regmode=DET \
regionfile=ah100050020sxs_region_SXS_det.reg sourcetype=FLATCIRCLE \
flatradius=10.0 rmffile=ah100050020_sxs_cl2_medium.rmf \
erange="0.5 17.0 0 0" outfile=ah100050020sxs_p0px1010_flat.arf \
numphoton=300000 minphoton=1 teldeffile=CALDB qefile=CALDB \
contamifile=CALDB obffile=CALDB fwfile=CALDB gatevalvefile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB backreffile=CALDB \
pcolreffile=CALDB scatterfile=CALDB
```

The following example produces an ARF for a beta-model distribution of core radius 1 arcminute radius, beta parameter 0.6, and maximum radius 10 arcminutes ('sourcetype=BETAMODEL, 'betapars="1.0,0.6,10.0" '), ah100050020sxs_p0px1010_beta.arf.

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxs_beta_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXS emapfile=ah100050020sxs_p0px1010.expo regmode=DET \
regionfile=ah100050020sxs_region_SXS_det.reg sourcetype=BETAMODEL \
betapars=”1.0,0.6,10.0” rmffile=ah100050020_sxs_cl2_medium.rmf \
erange="0.5 17.0 0 0" outfile=ah100050020sxs_p0px1010_beta.arf \
numphoton=300000 minphoton=1 teldeffile=CALDB qefile=CALDB \
contamifile=CALDB obffile=CALDB fwfile=CALDB gatevalvefile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB backreffile=CALDB \
pcolreffile=CALDB scatterfile=CALDB
```

The following example produces an ARF for an input 2-8 keV image, image.fits, ('sourcetype=image', 'imgfile=image.fits'), ah100050020sxs_p0px1010_image.arf. The third and fourth entries in the erange parameter correspond to the image bandpass (here, 2-8 keV).

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxs_image_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXS emapfile=ah100050020sxs_p0px1010.expo regmode=DET \
regionfile=ah100050020sxs_region_SXS_det.reg sourcetype=image \
imgfile=image.fits rmffile=ah100050020_sxs_cl2_medium.rmf \
erange="0.5 17.0 2.0 8.0" outfile=ah100050020sxs_p0px1010_image.arf \
numphoton=300000 minphoton=1 teldeffile=CALDB qefile=CALDB \
contamifile=CALDB obffile=CALDB fwfile=CALDB gatevalvefile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB backreffile=CALDB  \
pcolreffile=CALDB scatterfile=CALDB
```

5.7 Summary of tasks used in this chapter

Below is the list of all the tasks used or mentioned in this chapter. Readers are referred to the help files of these tasks for a more extensive description.

- aharfgen: Make an ancillary response function (ARF) file for the SXS or SXI, or a response matrix (RSP) file for the HXI
- ahexpmap: Generate an exposure map for HXI, SXI, and SXS, or a flat field image for SXI and SXS
- ahfilter: Generate an EHK file and a MKF filter file
- ahgtigen: Create and/or merge GTI files
- ahmkregion: Create regions files showing the field of view of all Hitomi instruments
- ahpipeline: HXI/SGD/SXS/SXI reprocessing tool
- ahscreen: Screen science event file
- coordevt: Convert events from one coordinate system to another
- fkeyprint: Print the specified keyword(s) in the headers of a list of input FITS files
- ftcopy: Copy the contents of a FITS file to a new file
- fthedit: Edit one or more header keywords in an input HDU
- ftselect: Copy selected rows from the input table to a new output table
- mxsgti: Create MXS GTI files
- sxsanticopi: Assign PHA and PI columns to the SXS anticoincidence events
- sxsbranch: Compute rates, branching ratios, and effective exposure times for each SXS event grade for an input real or simulated file, or based on a count rate
- sxsflagpix: Flag SXS events for antico and MXS event coincidence, temporal proximity, and crosstalk

- sxsgain: Calculate the time-dependent energy correction for SXS events from comparison with known calibration lines
- sxsmkfrmf: Create an SXS RMF file for a selection of pixels and grades
- sxsnxbgen: Create a Non-X-Ray Background (NXB) spectrum for SXS
- sxspha2pi: Calculate the PI for the SXS event files
- sxspipeline: SXS reprocessing tool
- sxsseccor: Correct PHA for secondary events
- sxssecid: Associate SXS secondary events to the primary and allow to recalculate event grades
- xselect: Extract image or spectrum for further analysis

6 SOFT X-RAY IMAGER (SXI) DATA ANALYSIS

6.1 Introduction

The SXI is an array of four X-ray CCDs at the focus of one of the two Hitomi Soft X-ray Telescopes (SXT-I). In their 2 x 2 layout, the detectors image a wide field (38 x 38 arcmin) with moderate energy resolution (E/ΔE ~40 @ 6 keV) in the soft X-ray band (0.4-12 keV), with a small gap of ~40 arcsec between the chips. Each detector is a p-channel, backside-illuminated (BI) CCD with an integrated framestore region, similar in operation to previous generations of X-ray CCDs, including the XIS aboard *Suzaku*. X-rays and particles are detected as charge clouds in the CCD lattice over the course of an exposure (4 sec in full window mode), and then transferred to the frame store region, where they are gradually read out and processed by the on-board electronics into events. The charge or "pulse-height" of each pixel in each event is altered by effects in the detectors and electronics, and these effects must be accounted for when calibrating and screening SXI events, extracting data products, and constructing response files.

6.2 Cleaned Event File Content

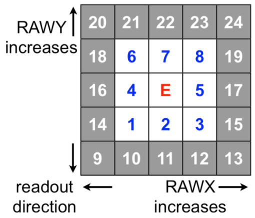
The files included in the standard data download in the directory sxi/event_uf were processed through the pipeline and calibrated accordingly. The primary calibration steps are summarized in Table 6.1. These were applied to the first FITS files, which were converted from Hitomi telemetry, and had time assigned and various keywords set.

Calibration Item	**Tool**	**Comments**
Calculate GTI	`sximodegti`	Generates GTIs for different observing modes
Assign coordinates to events	`coordevt`	For events: up to and including sky coordinates
Assign coordinates to hot pixel list	`coordevt`	For hot pixel list: up to and including DET coordinates
Merge pulse heights	`sxiphas`	Merge inner 3x3 and outer 5x5 pulse heights
Flag pixels	`sxiflagpix`	Flag SXI events for bad and hot pixels, CCD boundaries, calibration source region
Correct energy scale and assign PI (step 1)	`sxipi`	Applies corrections for even/odd, charge trail, CTI (without grade dependence), and calculates initial grade and summed PHA
Correct energy scale and assign PI (step 2)	`sxipi`	Applies grade-dependent CTI correction, calculates final grade and summed PHA, and applies gain correction to assign PI
Identify flickering pixels	`searchflickpix`	Only events with good GRADE and STATUS are used
Assign coordinates to flickering pixel list	`coordevt`	For flickering pixel list: up to and including DET coordinates
Flag pixels	`sxiflagpix`	Final event flagging including flickering pixels

Table 6.1: Primary calibration steps

Data taken in 1/8 window or 1/8 window+area discrimination (DATAMODE = WINDOW2) and/or burst mode (DATAMODE = WINDOW1BURST or WINDOW2BURST), or full window + 0.1 sec burst mode (Crab Nebula data only; DATAMODE = WINDOW1BURST2) have separate event lists for pairs of CCDs, with CCD1+CCD2 (keyword DETNAM = CCD12) in one file and CCD3+CCD4 (DETNAM = CCD34) in another. Because of this, and because the timing and response parameters are different, the calibration steps are separately applied to each event list, and data products such as spectra must be independently extracted and analyzed. When all four CCDs are operated in normal full window mode (DATAMODE = WINDOW1), the data obtained are included in one event file with DETNAM = CCD.

The columns in the SXI event file, which are relevant for screening, selection, and event extraction are listed in Table 6.2. To discriminate between real X-ray events and particle events, CCD detectors record the pulse heights in an island of pixels surrounding the event peak. The GRADE column in the event file provides a numerical representation of the size and shape of the charge cloud in the CCD lattice. The GRADE definition for the SXI is identical to that used in the *Suzaku*/XIS, and it delineates broad categories of events depending on which pixels in a 5x5 region centered on the event peak have a pulse height above the split threshold. This GRADE information is also used to sum a subset of the (corrected) individual pixel pulse-heights to obtain the total PI value, which is directly proportional to energy. The pulse-height recorded in a pixel for X-ray events that fall on an edge of a segment or in defective parts of the CCD are unreliable. The STATUS column encodes this information for each event and can be used to filter out X-ray events that may have compromised PI values, as well as non X-ray events.

Column	**Description**	**Range**
GRADE	Grade calculated from the 5x5 pixel pulse heights (same scheme as Suzaku XIS). ‘Left’ and ‘right’ refer to the figure below, where the event peak is at pixel ‘E’:  RAWY increases 20 21 22 23 24 18 6 7 8 19 16 4 E 5 17 14 1 2 3 15 9 10 11 12 13 readout direction RAWX increases GRADE=0: single pixel GRADE=1: corner split GRADE=2: vertical split GRADE=3: left split GRADE=4: right split GRADE=5: vertical, left, or right split with a corner GRADE=6: 2x2 box split with a low corner pulse-height GRADE=7: three pixel vertical or horizontal split with an outer 5x5 pixel above threshold GRADE=8: unused GRADE=9: three pixel vertical or horizontal split with no outer 5x5 pixel above threshold GRADE=10: same as GRADE 2, 4, 5, or 6 with an outer 5x5 pixel above threshold	0-11

	GRADE=11: 2x2 box split with a high corner pulse-height	
PI	Linearized Energy Channel	0-4095 1 PI chan=6 eV
STATUS	48 bit Event Status Flags (37 in use):	
	STATUS[1]: Bad event based on all other status flags	0-1
	STATUS[2]: Inside the calibration source region	0-1
	STATUS[3]: Out of CCD	0-1
	STATUS[4]: Out of window	0-1
	STATUS[5]: Out of area discrimination	0-1
	STATUS[6]: Charge Injection (CI) row	0-1
	STATUS[7]: Bad pixel from CALDB	0-1
	STATUS[8]: Bad column from CALDB	0-1
	STATUS[9]: Hot pixel from pre-pipeline	0-1
	STATUS[10]: Flickering pixel	0-1
	STATUS[11]: CCD boundary	0-1
	STATUS[12]: Window boundary	0-1
	STATUS[13]: Segment boundary	0-1
	STATUS[14]: Area discrimination boundary	0-1
	STATUS[15]: At least one 3x3 surrounding pixel has bad status	0-1
	STATUS[16]: CI trailing row	0-1
	STATUS[17]: CI preceding row	0-1
	STATUS[18]: Preceding/following of bad column	0-1
	STATUS[19]: Neighbor of bad/hot pixel and bad column	0-1
	STATUS[20]: Neighbor of flickering pixel	0-1
	STATUS[21]: Neighbor of preceding/following of bad column	0-1
	STATUS[22]: Neighbor of CCD/window boundary	0-1
	STATUS[23]: Neighbor of segment boundary	0-1
	STATUS[24]: 3x3 info is present but 5x5 is absent	0-1
	STATUS[25]: 3x3 is absent	0-1
	STATUS[26]: PHAS[0] < event threshold	0-1
	STATUS[27]: Video temperature is out of range	0-1
	STATUS[28]: Lack of video temp HK at time close to the event	0-1
	STATUS[29]: Correction value is negative	0-1
	STATUS[30]: Null value by correction process	0-1
	STATUS[31]: 1st trailing row of the CI rows	0-1
	STATUS[32]: 1st preceding row of the CI rows	0-1
	STATUS[33]: 2nd trailing row of the CI rows	0-1
	STATUS[34]: 2nd preceding row of the CI rows	0-1
	STATUS[35]: 3rd trailing row of the CI rows	0-1
	STATUS[36]: 3rd preceding row of the CI rows	0-1
	STATUS[37]: Cosmic ray echo pixel	0-1

Table 6.2: SXI event file columns relevant for calibration, screening and filtering

The cleaned event files in the sxi/event_cl directory are derived from the unfiltered event files in sxi/event_uf using two broad classes of screening: event-by-event and by good-time intervals (GTI). The event-by-event screening removes events that are more likely due to particles rather than X-rays, and events that might have compromised PI values as described above. The GTI screening applies criteria for nominal instrument status based on housekeeping data encapsulated in the makefilter (MKF) file in the auxil directory. GTI based on orbit- and pointing-derived criteria (stable pointing, unblocked by the earth, outside of regions of high background) encapsulated in the extended housekeeping (EHK) file in the auxil directory are also applied. The event-based and GTI-based filtering criteria are detailed in Table 6.3. The current screening criterion is based on an approximate characterization of the SAA, which could be improved in the future by utilizing the variation of the rate of detected SXS antico events in some energy range. The GTI constructed from ehk and mkf files are remade in the course of re-screening the data – either automatically (if using the *ahpipeline* script), or manually.

Type	File	Criterion	Comments
event-based	event_uf/OBSIDs xi_uf.evt	GRADE==0,2,3,4,6	X-ray event with reliable PI
		STATUS[3]==b0	In the good detector area
		STATUS[4]==b0	In the good detector area
		STATUS[5]==b0	In the good detector area
		STATUS[6]==b0	Not in charge injection (CI) row
		STATUS[7]==b0	Not in a bad pixel
		STATUS[8]==b0	Not in a bad column
		STATUS[9]==b0	Not in a hot pixel
		STATUS[10]==b0	Not in a flickering pixel
		STATUS[11]==b0	Not on a CCD boundary
		STATUS[12]==b0	Not on a window boundary
		STATUS[16]==b0	Not in a row trailing a CI row
		STATUS[17]==b0	Not in a row preceding a CI row
		STATUS[18]==b0	Not preceding/following bad col.
		STATUS[19]==b0	Not neighbor of bad pixel
		STATUS[20]==b0	Not neighbor of flickering pixel
		STATUS[25]==b0	All 3x3 PHAS present
		STATUS[26]==b0	Central PH > 0
		STATUS[27]==b0	Video temperature in range
		STATUS[28]==b0	Video temperature HK is present
		STATUS[30]==b0	Successful sxipi correction
		STATUS[37]==b0	Not cosmic ray echo pixel
		PROC_STATUS[1]==b0	
		PROC_STATUS[2]==b0	
GTI-based	auxil/OBSID.mkf	(SXI_USR_CCD1_OBS_MODE == 1 || SXI_USR_CCD2_OBS_MODE == 1 || SXI_USR_CCD3_OBS_MODE == 1 || SXI_USR_CCD4_OBS_MODE == 1)	Nominal instrument status
GTI-based	auxil/OBSID.ehk	ANG_DIST<1.5	Pointing within 1.5 arcmin of
		SAA_SXI==0	mean Satellite outside SAA
		T_SAA_SXS>277	Time since SAA passage > 277 s
		ELV>5	Pointing > 5 deg above Earth
		DYE_ELV>20	Pointing > 20 deg above Earth sunlit limb
		MZDYE_ELV>MZNTE_	Day earth elevation of minus-z

		ELV||MZDYE_ELV>20	direction > minimum of 20 deg and night earth elevation of minus-z direction
GTI-based	auxil/OBSIDsxi_tel.gti	inside GTITEL	Excludes times of telemetry saturation
GTI-based	OBSID_gen.gti	inside GTIPOINT and GTIATT	Excludes slew and bad attitude intervals

Table 6.3: Default of event-based and GTI-based filtering criteria.

The processing and screening summarized in this section results in an event file suitable for extraction of data products. Before proceeding, the user should ascertain whether any SXI calibration files relevant to the time of the observation have changed since the observation was processed (see the explanation of CALDBVER in Section 3.2) by checking http://heasarc.gsfc.nasa.gov/docs/hitomi/calib/ and the list of calibration files used in the SXI pipeline shown in Table 6.4. Reprocessing of SXI events should be performed first if (1) there are any such changes to the relevant CALDB files used in processing and screening; or (2) any screening criteria different from the standard are applied (see Table 6.3).

6.3 Reprocessing Events

The calibration files used in SXI pipeline processing are listed in Table 6.4.

Tool	Calibration File	Description
sxiflagpix	ah_sxi_badpix_yyyymmddvNNN.fits ah_sxi_mask_yyyymmddvNNN.fits	Bad pixel file Instrument mask file
sxipi	ah_sxi_vtevnodd_yyyymmddvNNN.fits ah_sxi_chtrail_yyyymmddvNNN.fits ah_sxi_cti_yyyymmddvNNN.fits ah_sxi_spth_yyyymmddvNNN.fits ah_sxi_gain_yyyymmddvNNN.fits ah_sxi_pattern_yyyymmddvNNN.fits	Even/odd column video board gain file Charge trail file Charge transfer inefficiency (CTI) file Split threshold file PHA->PI gain correction file Initial grade hit pattern file
ahfilter	ah_gen_mkfconf_yyyymmddvNNN.fits	Makefilter file for screening
ahgtigen	ah_gen_select_yyyymmddvNNN.fits	Selection file for screening

Table 6.4: SXI CALDB files used in calibration and screening.

(1) If the *ahpipeline* script was run, the event files for all of the Hitomi instruments were reprocessed, and there is no need to conduct an SXI-specific reprocessing. To reprocess only the SXI event files, run the *sxipipeline* script script (equivalent to running *ahpipeline* with 'instrume=SXI' instead of 'instrume=ALL') as follows. From the directory above where the original data is located, and for an observation with OBSID ah100050020, issue the command

```
%>sxipipeline indir=data/100050020 outdir=data/100050020_repro_sxi \
steminputs=ah100050020 stemoutputs=DEFAULT entry_stage=1 \
exit_stage=2 attitude=data/100050020/auxil/ah100050020.att.gz \
orbit=data/100050020/auxil/ah100050020.orb.gz \
extended_housekeeping=data/100050020_repro_sxi/ah100050020.ehk \
makefilter=data/100050020_repro_sxi/ah100050020.mkf \
obsgti=data/100050020/auxil/ah100050020_gen.gti.gz \
housekeeping=data/100050020/sxi/hk/ah100050020sxi_a0.hk.gz
```

to reconstruct the unfiltered and cleaned event files and place them in the repro_sxi subdirectory. To clean the data only, change the 'entry_stage' parameter from 1 to 2; to recalibrate the data only, change the 'exit_stage' parameter from 2 to 1.

(2) The method for applying non-standard screening criteria depends on the nature of the difference. A stricter cut on columns in the extended housekeeping (EHK) file, makefilter (MKF) file, or event file can be applied to the cleaned data using *xselect*, e.g.,

```
xsel:HITOMI-SXI-WINDOW1 > read hk hkfiles=../auxil/ah100050020.ehk \
expand=yes
xsel:HITOMI-SXI-WINDOW1 > select hk "DYE_ELV>30"
```

imposes a stricter limit on the angle from the bright Earth limb, while

```
xsel:HITOMI-SXI-WINDOW1 > filter column "STATUS[2]==b0"
```

excludes any events that fall in the calibration source regions.

A more general rescreening requires a multi-step reprocessing of the unfiltered event files, which is explained in detail in Appendix A.

6.4 Extracting Products

This section summarizes the steps needed for the extraction of data products – images, spectra, and lightcurves – including additional screening, region and grade selection, and caveats to watch out for. See Section 9 for a summary of the parallel analysis steps for all Hitomi instruments. These steps are applied to "NORMAL" event data (of the form ahXXXXXXXXXsxi_PRmmmmmmb**0**_cl.evt), and not the "MINUS-Z DAY EARTH (MZDYE) DATA" (of the form ahXXXXXXXXXsxi_PRmmmmmmb**1**_cl.evt), which is affected by the SXI light leak. However, the identical steps may be applied to the MZDYE DATA.

6.4.1 Additional screening of events

1) Filtering times of high COR

To reduce the non-X-ray background (NXB), a time filter can be applied to eliminate any data associated with high COR (Cut-Off Rigidity). The exact COR cut-off depends on the particular observation and science goals, however a rule of thumb is that low surface brightness observations may benefit from a stricter minimum COR cutoff, while for bright point sources such a restriction is less important, and it reduces the effective exposure time. The COR information is contained in the EHK file in a number of columns; the recommended value to use is COR2, as in the following pair of *xselect* commands that eliminates time intervals associated with locations in the orbit where COR2>6. :

```
xsel:HITOMI-SXI-WINDOW1 > read hk hkfiles=../auxil/ah100050020.ehk \
expand=yes
xsel:HITOMI-SXI-WINDOW1 > select hk "COR2.GE.6"
```

The resulting filtered event file can be saved with the *xselect* commands:

```
xsel:HITOMI-SXI-WINDOW1 > extract events
xsel:HITOMI-SXI-WINDOW1 > save events
ah100050020sxi_p0100004b0_cl_cor.evt
```

2) Creating regions from an image

Spectral extraction regions are most easily created by hand from examination of an image in *ds9* or *ximage*. For the SXI, creating this image should be paired with a cut on energy to reduce the background. We suggest a first cut excluding all events outside the [0.5-8.0] keV range, or [83-1333] in PI since there are 6 eV per PI channel. Note that this energy range includes the bulk of flux from a typical X-ray point source such as an AGN or X-ray binary while excluding high-energy photons from the non-X-ray background. In *xselect*, the following commands create such an image:

```
xsel:HITOMI-SXI-WINDOW1 > filter pha cutoff 84 1333
xsel:HITOMI-SXI-WINDOW1 > extract image
xsel:HITOMI-SXI-WINDOW1 > save image g21p5_broadband_img.fits
```

Once the image has been created and displayed one can define the regions of scientific interest. For visualization, regions can be created using the task *ahmkregion* to overlay the different fields of view for the SXS or HXI on the SXI image (see Sections 4.4.1). The region can be defined in one of several coordinate systems (see Section 2.3.1). Users who want to use the exact same region for all detectors should use either SKY or FOC coordinates. Regions may span multiple CCDs in a single event list, and the missing region in the gap between CCDs is properly accounted for by the response files (using the WMAP for the RMF and the exposure map for the ARF). Calibration source emission can be removed either by excluding the calibration source region explicitly with a region file, or by using the command

```
xsel:HITOMI-SXI-WINDOW1 > filter STATUS[2]==b0
```

An example of selecting source and background regions for an observation of G21.5-0.9 is shown in Figure 6.1. The background region is constructed so as to exclude the Fe55 calibration source regions, as well as the edges of the detector to reduce number the out-of-time events from the calibration source.

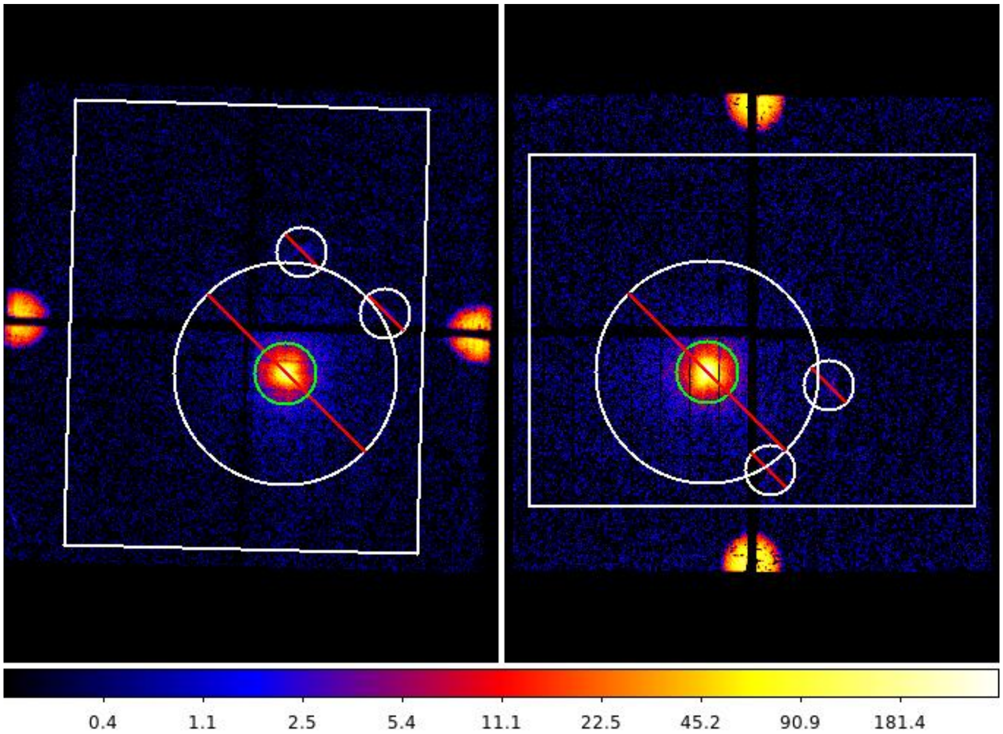


Figure 6.1: Examples of defining regions of interest . The left and right panels are given in the SKY and DET coordinates, respectively. The green and white circles represent the source (G21.5-0.9) and background extraction regions, respectively.

3) Caveats for high count rate sources

For bright point sources, the effect of pile-up should be estimated using the tool *pileest* and its effect mitigated by choice of extraction region. The output of *pileest* is an image that maps the estimated pile-up fraction, and the tool is run on the command line as follows:

```
pileest eventfile=ah100050020sxi_p1100004b0_uf.evt \
outmap=pilest_img.fits inreg=NONE outreg=pilest_exclude.reg \
alpha=0.5 maxpilefrac=0.08 interactive=no \
gtifile=NONE contpfrac=yes pfrac_vals="0.01,0.02,0.05,0.1" \
plotimage=yes plotdevice="pileest.ps/cps"
```

The output region file from this command, pilest_exclude.reg, is defined as the smallest circle required to exclude pixels with fractional pileup>maxpilefrac (in this case 8%) and may be used as the spectral extraction region. If interactive=yes instead, then the user can create such a region using a displayed image of the pile-up fraction. If a region file is input using the 'inreg' parameter, then the parameters of the XSPEC pileup model corresponding to this region are output. Note that this usage requires that the spectrum be extracted from this same region.

6.4.2 Examples of Extracting Products

The barycentric correction may be applied prior to extraction of data products using the *barycen* task, which requires an orbit file and source coordinates as input, as shown in the following example:

```
%>barycen infile=ah100050020sxi_p0100004b0_cl.evt.gz \
outfile=ah100050020sxi_p0100004b0_cl_barycor.evt \
orbfile=ah100050020.orb.gz ra=278.38824  dec=-10.570683 orbext=ORBIT
```

Note that the 'ra' and 'dec' parameters should be set to the true average pointing direction. This should be set to the RA_NOM and DEC_NOM keywords in the header for cases where the attitude is correct, or may be approximated by the object coordinates in cases where the source is known to be on-axis.

A) Point-Like Source

The *xselect* session below shows an example of extracting data products from the observation of a point-like source. Here we use the existing data of G21.5-0.9, which includes extended emission with a diameter of ~5', in addition to a small pulsar-wind nebula, but is treated here as point-like. First an image is extracted from a restricted energy band, and source and background regions are made by hand (see Figure 6.1). The background region is used to extract a light curve to check for time variability. Then background and source spectra are extracted.

1) Read in the SXI event list data.

The cleaned event file is used here.

```
xsel:SUZAKU >read events ah100050020sxi_p0100004b0_cl.evt
```

2) Extract an image to define source and background regions.

A binned image in the 0.5-8 keV bandpass is used to define source and background extraction regions.

```
xsel:HITOMI-SXI-WINDOW1 > set xybinsize 4
xsel:HITOMI-SXI-WINDOW1 > filter pha_cutoff 84 1333
xsel:HITOMI-SXI-WINDOW1 > extract image
xsel:HITOMI-SXI-WINDOW1 > save image g21p5_sxi.img clobberit=yes
xsel:HITOMI-SXI-WINDOW1 > plot image
```

The extracted image, and the source and background regions, are shown in Figure 6.1 above.

3) Extract source and background spectra

The steps here include removing the energy cut, filtering on region, and resetting the WMAP bin size to reduce the file size. The WMAP bin columns are properly set to DET coordinates by default.

```
xsel:HITOMI-SXI-WINDOW1 > clear pha_cutoff
xsel:HITOMI-SXI-WINDOW1 > set wmapbinsize 4
xsel:HITOMI-SXI-WINDOW1 > filter region g21p5_src.reg
xsel:HITOMI-SXI-WINDOW1 > extract spectrum
xsel:HITOMI-SXI-WINDOW1 > save spectrum spec_src clobberit=yes
xsel:HITOMI-SXI-WINDOW1 > clear region
xsel:HITOMI-SXI-WINDOW1 > filter region g21p5_bg.reg
xsel:HITOMI-SXI-WINDOW1 > extract spectrum
xsel:HITOMI-SXI-WINDOW1 > save spectrum spec_bg clobberit=yes
```

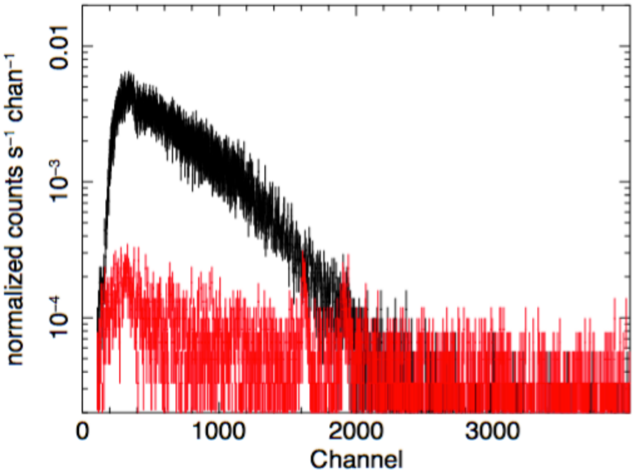


Figure 6.2: PI channel spectra of source (black) and background (red) from the G21.5-0.9 observation.

B) Extended Source

The *xselect* session below shows an example of extracting data products from an observation of an extended source such as a galaxy cluster. First an image is extracted from a restricted energy band, and source and background regions made by hand. The background region is used to extract a light curve to look for time variability. Background and source spectra are extracted.

1) Read in the SXI event list data.

```
xsel:SUZAKU >read events cluster_with_pts.evt
```

2) Extract an image to define source and background regions.

A binned image in the 0.5-8 keV bandpass is again used to define source and background extraction regions.

```
xsel:HITOMI-SXI-WINDOW1 > set xybinsize 4
xsel:HITOMI-SXI-WINDOW1 > filter pha_cutoff 84 1333
```

```
xsel:HITOMI-SXI-WINDOW1 > extract image
xsel:HITOMI-SXI-WINDOW1 > save image cluster_with_pts.img
clobberit=yes
xsel:HITOMI-SXI-WINDOW1 > plot image
```

Figure 6.3 shows the extracted image along with regions defined for the source (left) and background (right). The white regions are inclusion regions, while the green circles with red lines are to be excluded. Note that the calibration sources, illuminating a corner of each CCD, have been excluded by hand. The edges of the detector may be excluded as well, as was done in the previous example.

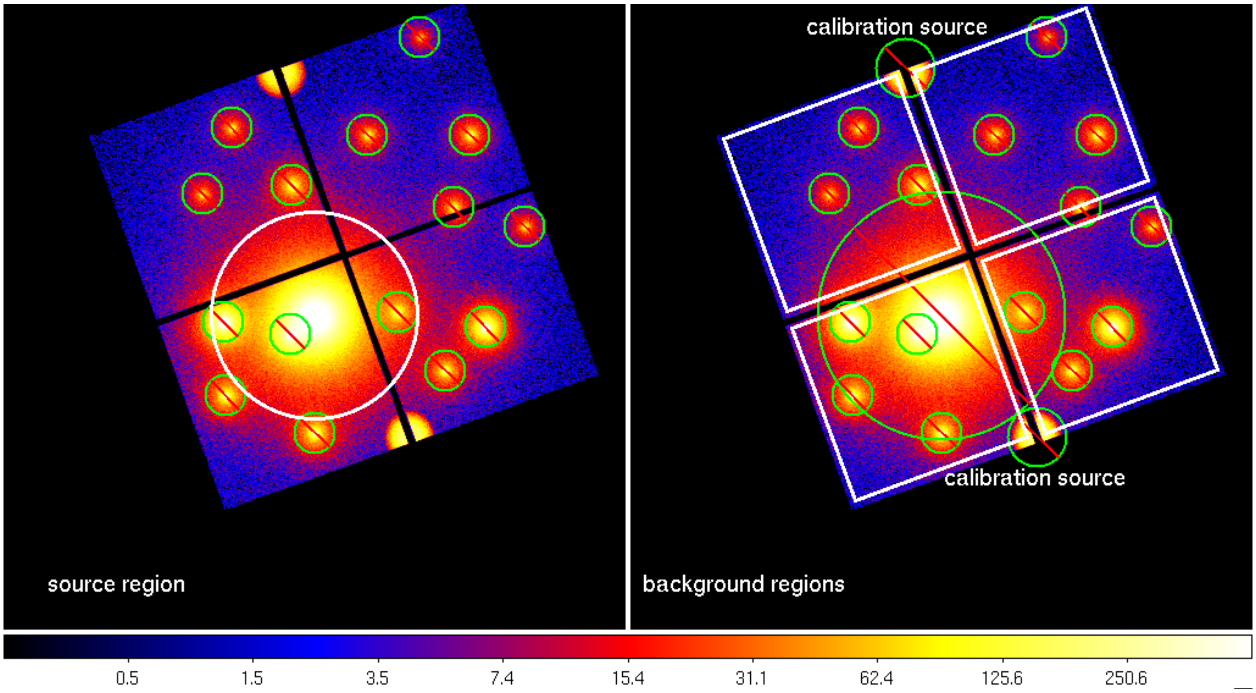


Figure 6.3: Extracted region with point sources and calibration sources excluded.

3) Extract a light curve from the background to look for time variability.

The background region file and an energy cut of 12-24 keV are used to extract the non-X-ray background lightcurve. The curve is shown in Figure 6.4.

```
xsel:HITOMI-SXI-WINDOW1 > filter region
cluster_with_pts_sxi_wcs_bg.reg
xsel:HITOMI-SXI-WINDOW1 > filter pha_cutoff 2000 4000
xsel:HITOMI-SXI-WINDOW1 > set binsize 300
xsel:HITOMI-SXI-WINDOW1 > extract curve
xsel:HITOMI-SXI-WINDOW1 > plot curve
```

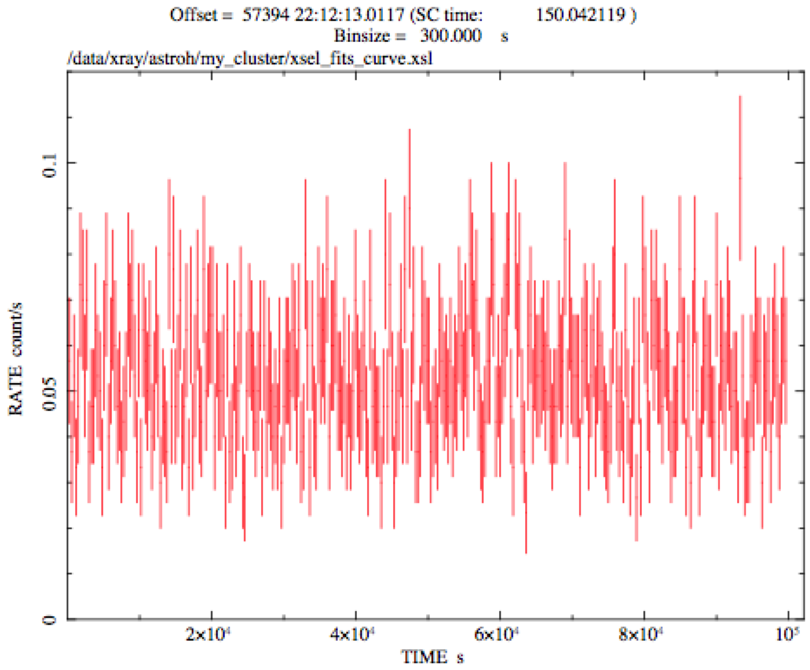


Figure 6.4: Plot of the output light curve from the background, in the 12-24 keV band.

4) Extract a background spectrum.

Since the background light curve is constant, the background spectrum is extracted after removing the energy cut, filtering on the background region, and resetting the WMAP bin size to reduce the file size.

```
xsel:HITOMI-SXI-WINDOW1 > clear pha_cutoff
xsel:HITOMI-SXI-WINDOW1 > set wmapbinsize 4
xsel:HITOMI-SXI-WINDOW1 > extract spectrum
xsel:HITOMI-SXI-WINDOW1 > save spectrum cluster_bg clobberit=yes
```

5) Extract the source (cluster) spectrum.

The source spectrum is extracted after resetting the region filter, and shown in Figure 6.5. Also shown are images of the WMAP for both the source and background extraction. Since the WMAP is in DET coordinates and binned by a factor of 4, it appears rotated and smaller compared to the images in SKY shown above.

```
xsel:HITOMI-SXI-WINDOW1 > clear region all
xsel:HITOMI-SXI-WINDOW1 > filter region cluster_with_pts_sxi_wcs.reg
xsel:HITOMI-SXI-WINDOW1 > extract spectrum
xsel:HITOMI-SXI-WINDOW1 > save spectrum cluster_bg clobberit=yes
```

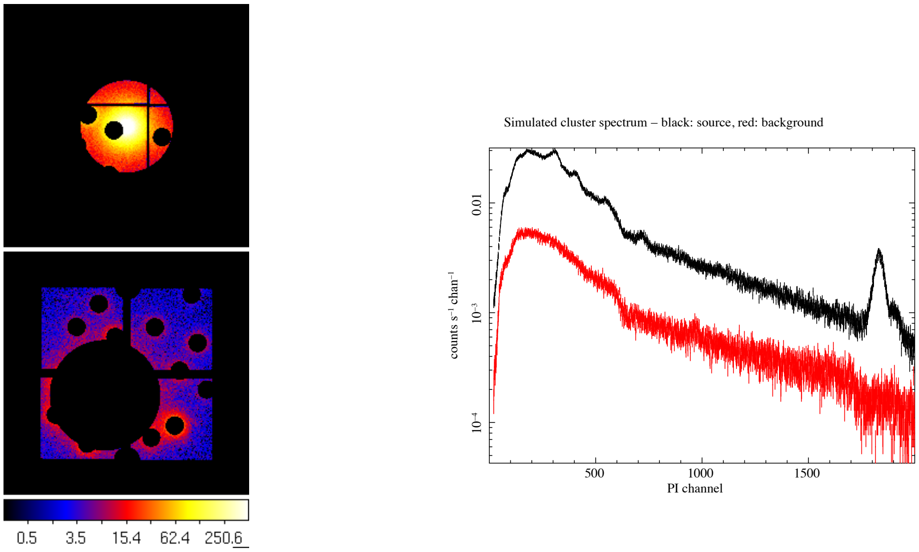


Figure 6.5 (Left): Images of the WMAP for both the source (top) and background extraction (bottom). (Right): Count spectrum of the simulated cluster.

6.5 Background estimation

Both external astrophysical and internal non-X-ray (NXB) components contribute to the total background associated with any observation. The local background, including the NXB, may be estimated using a spectrum extracted identically to the source region but from a separate off-source region. This is usually possible for a point-source observed with SXI. For an extended source which fills the field of view (such as a nearby cluster of galaxies), a separate blank-sky pointing can be used. Due to the effects of telescope vignetting and variation of the NXB across an SXI CCD and from detector to detector, SXI spectral analysis should include *subtracting* the NXB background and *simultaneously fitting* the X-ray background and source spectra, i.e. modeling the X-ray background. The NXB spectrum may be estimated using the task *sxinxbgen*. This task requires input source and NXB event files, input source and NXB EHK files, and a region file. The NXB filtering must match that used to derive the source spectrum. Returning to the example in Section 6.4.1 the command to create the NXB spectrum is:

```
sxinxbgen infile=cluster_with_pts.evt \
ehkfile=../auxil/ah000506304.ehk.gz \
regfile=cluster_with_pts_sxi_wcs.reg \
innnxbfile=nxb.fits innxbehk=nxb.ehk \
outpifile=ah000506287sxs_p0px0000_nxb.pi \
sortcol="COR2" sortbin="0,4,5,6,7,8,9,10,11,12,13,99" \
apply_sxipi=yes
```

where the ‘sortcol’ and ‘sortbin’ parameters are chosen to match the COR2 filtering. Note that we performed a COR2 filter on the input event file in Section 6.4.1, excluding any events during times of COR2 < 6. By specifying ‘apply_sxipi=yes’, *sxinxbgen* reprocesses the NXB event list to apply the latest calibration. *Note that, because of the short HITOMI lifetime, the sxinxbgen* *tool is not relevant. Standard SXI background spectra based on observations with no source may be available in the future.*

6.6 ARF and RMF generation

The following example illustrates the steps required to make an ARF and RMF for an SXI spectrum. The spectrum itself is required as well as the region file from which the spectrum is extracted.

6.6.1 RMF generation with *sxirmf*

The following comman produces an RMF for the spectrum ah100050020sxi_p0100004b0_cl.pi.

```
%>sxirmf infile=ah100050020sxi_p0100004b0_cl.pi /
outfile=ah100050020sxi_p0100004b0_cl.rmf
```

6.6.2 ARF file generation using *ahexpmap* and *aharfgen*

The first step in creating an ARF file is to run *ahexpmap*, as in the following example. Some of the hidden parameters are also shown for completeness. The input parameters include the extended HK file for this observation (parameter 'ehkfile'), the cleaned event file that provides the GTI (parameter 'gtifile'), the flickering pixel file (parameter 'pixgtifile'), the bad pixel image file (parameter 'badimgfile'), and the attitude binning parameters 'delta' and 'numphi'

```
%>ahexpmap ehkfile=ah100050020.ehk.gz \
gtifile=ah100050020sxi_p0100004b0_cl.evt instrume=SXI \
badimgfile=ah100050020sxi_p0100004b0.bimg.gz \
pixgtifile=ah100050020sxi_a0100004b0.fpix.gz \
outfile=ah100050020sxi_p0100004b0.expo outmaptype=EXPOSURE \
delta=0.25 numphi=4 stopsys=SKY instmap=CALDB qefile=CALDB \
contamifile=CALDB vigfile=CALDB obffile=CALDB fwfile=CALDB \
gvfile=CALDB maskcalsrc=yes fwtype=DEFAULT specmode=MONO \
specfile=spec.fits specform=FITS energy=1.5 evperchan=DEFAULT \
abund=1 cols=0 covfac=1
```

If the attitude is stable, or inaccurate based on a comparison of the centroid of the SXI image with the source coordinates, one can run *ahexpmap* with 'numphi=1' and a large value of 'delta' (~100) and so that there is a single attitude bin in the output histogram. Generally the 'FRACTION' column in the 'OFFAXISHIST' extension of the exposure map should be examined, and the value of 'delta' adjusted to eliminate entries with values <<1 to save computing time when constructing the ARF.

The next step is to run *aharfgen,* a script that internally calls *ahsxtarfgen*, to create the ARF file ah100050020sxi_p0px1010_point.arf. The following example produces an ARF for a point source (parameter 'sourcetype=POINT'). The input parameters include the source coordinates (parameters 'source_ra' and 'source_dec'), the exposure map produced in the previous step (parameter 'emapfile'), the extraction region in DET coordinates (parameter 'regionfile'), and the energy range (parameter ‘erange’). If the attitude is suspect, the centroid of the source SXI image should be used in place of the true source coordinates. The expected runtime for the set of parameters below is ~30-40 minutes per attitude bin, and the value of ‘numphoton’ should be adjusted accordingly. An on-axis ARF may be generated by constructing an exposure may with a single attitude bin, and setting source_ra’ and ‘source_dec’ to the values in the RANOMXP and DECNOMXP column.

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxi_ptsrc_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXI dattfile=NONE filtoffsetfile=NONE \
emapfile=ah100050020sxi_p0100004b0.expo \
gatevalvefile=CALDB sampling=1 regmode=SKY \
regionfile=region_SXI_100050020_sky.reg sourcetype=POINT \
rmffile=ah100050020sxi_p0100004b0_cl.rmf erange="0.3 17.0 0 0" \
outfile=ah100050020sxi_p0100004b0_point.arf numphoton=300000 \
minphoton=1 teldeffile=CALDB qefile=CALDB contamifile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB backreffile=CALDB \
pcolreffile=CALDB scatterfile=CALDB imgfile=NONE
```

The following example produces an ARF for a uniform circular distribution of 10 arcminute radius (‘sourcetype=POINT’, ‘flatradius=10.0’), ah100050020sxi_p0100004b0_flat.arf.

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxi_flat_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXI dattfile=NONE filtoffsetfile=NONE\
emapfile=ah100050020sxi_p0100004b0.expo \
gatevalvefile=CALDB sampling=1 regmode=SKY \
regionfile=region_SXI_100050020_sky.reg sourcetype=FLATCIRCLE  \
flatradius=10.0 rmffile=ah100050020sxi_p0100004b0_cl.rmf \
erange="0.3 17.0 0 0" outfile=ah100050020sxi_p0100004b0_flat.arf \
numphoton=300000 minphoton=1 teldeffile=CALDB qefile=CALDB \
contamifile=CALDB onaxisffile=CALDB onaxiscfile=CALDB \
mirrorfile=CALDB obstructfile=CALDB frontreffile=CALDB \
backreffile=CALDB pcolreffile=CALDB scatterfile=CALDB \
imgfile=NONE
```

The following example produces an ARF for a beta-model distribution of core radius 1 arcminute, beta parameter 0.6, and maximum radius 10 arcminutes (‘sourcetype=BETAMODEL, ‘betapars=”1.0,0.6,10.0” ’), ah100050020sxi_p0px1010_beta.arf.

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxi_beta_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXI dattfile=NONE filtoffsetfile=NONE\
emapfile=ah100050020sxi_p0100004b0.expo \
gatevalvefile=CALDB sampling=1 regmode=SKY \
regionfile=region_SXI_100050020_sky.reg sourcetype=BETAMODEL \
betapars=”1.0,0.6,10.0” rmffile=ah100050020sxi_p0100004b0_cl.rmf \
erange="0.3 17.0 0 0" outfile=ah100050020sxi_p0100004b0_beta.arf \
numphoton=300000 minphoton=1 teldeffile=CALDB qefile=CALDB \
contamifile=CALDB onaxisffile=CALDB onaxiscfile=CALDB \
mirrorfile=CALDB obstructfile=CALDB frontreffile=CALDB \
backreffile=CALDB pcolreffile=CALDB scatterfile=CALDB \
imgfile=NONE
```

The following example produces an ARF for an input 2-8 keV image, image.fits arcminutes ('sourcetype=image', 'imgfile=image.fits'), ah100050020sxi_p0px1010_image.arf. The third and fourth entries in the erange parameter correspond to the image bandpass (here, 2-8 keV).

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxi_image_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXI dattfile=NONE filtoffsetfile=NONE\
emapfile=ah100050020sxi_p0100004b0.expo \
gatevalvefile=CALDB sampling=1 regmode=SKY \
regionfile=region_SXI_100050020_sky.reg sourcetype=image \
imgfile=image.fits rmffile=ah100050020sxi_p0100004b0_cl.rmf \
erange="0.3 17.0 0 0" outfile=ah100050020sxi_p0100004b0_image.arf \
numphoton=300000 minphoton=1 teldeffile=CALDB qefile=CALDB \
contamifile=CALDB onaxisffile=CALDB onaxiscfile=CALDB \
mirrorfile=CALDB obstructfile=CALDB frontreffile=CALDB \
backreffile=CALDB pcolreffile=CALDB scatterfile=CALDB
```

6.7 Summary of the tasks of this section

Below is the list of all the tasks used or mentioned in this chapter. Readers are referred to the help files of these tasks for a more extensive description.

- aharfgen: Make an ancillary response function (ARF) file for the SXS or SXI, or a response matrix (RSP) file for the HXI
- ahexpmap: Generate an exposure map for HXI, SXI, and SXS, or a flat field image for SXI and SXS
- ahfilter: Generate an EHK file and a MKF filter file
- ahgtigen: Create and/or merge GTI files
- ahmkregion: Create region files showing the fields of view of all Hitomi instruments
- ahpipeline: HXI/SGD/SXS/SXI reprocessing tool
- barycen: Correct time to Solar System barycenter to prepare for light curve extraction
- coordevt: Convert events from one coordinate system to another
- pileest: Compute an image of the estimated pile-up in the defined region
- searchflickpix: Search for flickering pixels in event files from CCD-type detectors
- sxiflagpix: Flag SXI events for bad and hot pixels, CCD boundaries, calibration source regions
- sximodegti: Generate GTIs excluding dead time for each SXI observing mode
- sxinxbgen: Create a Non X-ray Background spectrum
- sxipi: Assign PI and grades for SXI events
- sxipipeline: SXI reprocessing tool
- sxirmf: Create an SXI RMF associated with a PHA file

- xselect: Extracts image or spectrum for further analysis

7 HARD X-RAY IMAGER (HXI)

7.1 Introduction

Hard photons collected by the Hard X-ray Telescopes are focused on the Hard X-ray Imager (HXI), a two dimensional imaging spectrometer working in the 5 to 80 keV X-ray energy band. Hitomi has 2 HXI+HXT pairs, each detector consisting of four layers of position sensitive Double-Sided Si Strip Detectors (DSSDs) and a single layer cadmium telluride (CdTe) semiconductor. A modularized active anti-coincidence shield made of Bismuth Germanate (BGO) scintillation crystals surrounds the Si and CdTe detectors. These crystals make a well-like deep structure narrowing the main detector aperture while the scintillation photons are read by attached Avalanche PhotoDiodes (APDs). The HXI focal position is located 232 mm above the surface of the HXI plate. The HXI detectors (HXI1 and HXI2) are rotated 22.5 degrees compared with the satellite coordinate in opposite directions, so they are rotated by 45 degrees with respect to one another. The field of view of an individual HXI detector is 9.17 arcmin x 9.17 arcmin square.

7.2 Cleaned Event File Content

The files included in the standard data download in the directory hxi/event_uf were processed through the pipeline and calibrated accordingly. The primary calibration steps are summarized in Table 7.1. These were applied to the first files that were converted from Hitomi telemetry, and had time assigned and various keywords set.

After unpacking the telemetry using *hxisgdsff* and assigning energy with *hxisgdpha*, each occurrence in the telemetered event file is reconstructed into a single photon interaction event using the tool *hxievtid.* This tool determines the signal validity of an occurrence and computes the energy of a photon detected in the HXI and the position of the first photon interaction. In the cleaned event file, each row corresponds to a reconstructed event and the columns 'RAWX', 'RAWY', ‘LAYER’, and 'PI' contain, respectively, the reconstructed position, interaction layer, and energy of that event, along with additional columns containing information related to the number and the type of interactions. If the algorithm is unable to reconstruct a valid event, then the ‘PI’ column value is set to NULL and diagnostic information is written to the output file. These NULL ‘PI’ events are filtered out in the event cleaning.

Calibration Step	Tool	Comments
Derive Δ-attitude	cams2att	Run twice; once for each HXI unit
Unpack telemetry	hxisgdsff	Output data stored in variable-length arrays
Compute EPI	hxisgdpha	Also flags bad pixels
Reconstruct events	hxievtid	Derives position and total deposited energy for events that can be properly reconstructed.
Transform coordinates	coordevt	Transform from RAW to SKY coordinates.

Table 7.1: Primary calibration steps.

The columns that can be used for data filtering are listed in Table 7.2.

Column	Description	Values
EVTCAT	Event category	1=single hit in one layer; 2-5=hit in CdTe layer and in Si detector; 6-10=various reconstruction failure modes.
GOODBAD	Flag to indicate signal quality per side (10-element array):	0=good signal, 1= null signals, 2=signals below threshold, 3=no signals, 4=signals null and below threshold
LAYER	Layer location of the event	0-3 DSSD, ordered from top to bottom; 4=CdTe

Table 7.2: Columns used for additional filtering.

Cleaned event files are separated by HXI unit. There are two files for each unit, e.g. ah100050020hx1_p0camrec_cl.evt.gz and ah100050020hx1_p0camrecpse_cl.evt.gz indicate normal and pseudo-events for sequence 100050020 HXI1. Normal and pseudo-event files have the same format and the same columns; however, only the photon events are to be used for science analysis; the pseudo-events are used for deadtime correction.

The reconstructed unfiltered event files (e.g. ah100050020hx1_p0camrec_ufa.evt) in the hxi/event_uf directory are screened using the criteria given in Table 7.3 to produce the cleaned event files using two broad classes of screening, event-by-event and by good-time intervals (GTI). The GTI screening filters slew and bad attitude intervals, and intervals of saturated telemetry. The GTI files needed for this screening are already present in the auxil directory or hxi/event_uf directories. In addition, GTI corresponding to nominal instrument status based on housekeeping data encapsulated in the makefilter (mkf) file in the auxil directory, and GTI-based on orbit- and pointing-derived criteria (times of stable pointing, unblocked by the earth, outside of regions of high background) encapsulated in the extended housekeeping (ehk) file in the auxil directory are applied. The screening criteria are based on an expanded definition of the SAA. The GTI constructed from ehk and mkf files are remade in the course of re-screening the data – either automatically (if using the *ahpipeline* script), or manually.

Type	File	Criterion	Comments

event-based	event_uf/OBSIDhx1_uf.evt	FLAG_HITPAT == b0
		&& FLAG_FASTBGO == b0
		&& FLAG_SEU == b0
		&& FLAG_LCHK == b0
		&& FLAG_TRIGPAT[6] == 0
		&& FLAG_TRIGPAT[7] == 0
		&& FLAG_TRIGPAT[8] == 0
		&& EVTCAT<=5
		&& PROC_STATUS[1] = 0
		&& PROC_STATUS[2] = 0
		&&((LAYER==0&&(PI<300&&PI>=0))||(LAYER==1&&(PI<2048&&PI>=120))||(LAYER==2&&(PI<2048&&PI>=120))||(LAYER==3&&(PI<2048&&PI>=120))||(LAYER==4&&(PI<2048&&PI>=300)))
			No shield signals;
			No fast BGO signals;
			No single event upset;
			Length check no error;
			No forced trigger;
			No pseudo trigger;
			No calibration pulse;
			Event category shows valid event;
			Proc_status bits not set;
			PI cuts specific to detector layers

GTI-based	auxil/OBSID.mkf	HXI1_USR_DE_MODE == 1 && HXI1_USR_MIO2_MODE == 2 && HXI1_USR_CPMU_HV1_CAL >= 200.0 && HXI1_USR_CPMU_HV2_CAL >= 200.0 && HXI1_USR_APMU_HV1_CAL >= 200.0 && HXI1_USR_APMU_HV2_CAL >= 200.0	Observing mode for HXI Detector and MIO2; All HXIn CPMU & APMU HVn >= 200.0 V (HXIn=1,2; HVn=1,2)
GTI-based	auxil/OBSID.ehk	ANG_DIST<1.5 && SAA_HXI1==0 && T_SAA_HXI1>251 && TN_SAA_HXI1>251 && ELV>5 && DYE_ELV>20	Pointing <1.5 arcmin of mean Satellite outside SAA Time since last SAA passage > 251s Time until next SAA passage > 251s More than 5 degrees Earth elevation; More than 20 degrees above above Earth sunlit limb
GTI-based	auxil/OBSID_gen.gti	inside GTIPOINT and GTIATT	Excludes slew and bad attitude intervals
GTI-based	auxil/OBSIDhxi_tel.gti	inside GTITEL	Excludes times of telemetry saturation

Table 7.3: Default event-based and GTI-based filtering criteria for HXI1. Similar criteria apply to HXI2.

The processing and screening summarized in this section results in an event file suitable for extraction of data products. However, reprocessing of HXI events ought to be applied first if: (1) there are changes in any of the relevant CALDB files used in processing and screening (see explanation of CALDBVER in Section 3.2), or (2) screening criteria different from the standard are applied. The user should ascertain whether any HXI calibration files relevant to the time of the observation have changed since the observation was processed by checking http://heasarc.gsfc.nasa.gov/docs/hitomi/calib/ and the list of calibration files used in the HXI pipeline shown in Table 7.4.

7.3 Reprocessing Events

The calibration files used in HXI pipeline processing are listed in Table 7.4.

Tool	Calibration File	Description
cams2att	ah_cm1_teldef_yyyymmddv00n.fits ah_cm2_teldef_yyyymmddv00n.fits ah_hxn_teldef_yyyymmddv00n.fits ah_cms_tempxy_yyyymmddv00n.fits	CAMS1 TelDef CAMS2 TelDef HXI TelDef CAMS Temperature correction file
hxisgdsff	ah_hxi_remap_ yyyymmddv00n.fits	HXI readout remapping file
hxisgdpha	ah_hxn_gain_yyyymmddv00n.fits ah_hxn_badpix_yyyymmddv00n.fits	HXI PHA calibration gain function Flags for active/dead/noisy readout channels
hxievtid	ah_hxi_remap_ yyyymmddv00n.fits ah_hxi_line_yyyymmddv00n.fits ah_hxn_badpix_yyyymmddv00n.fits ah_hxn_enecut_yyyymmddv00n.fits	HXI readout remapping file HXI Fluorescence file HXI energy threshold file HXI energy cut file
coordevt	ah_hxn_teldef_yyyymmddv00n.fits	HXI TelDef

Table 7.4: Calibration database files used in Primary calibration steps.

To reprocess only the HXI event files, the *hxipipeline* script (equivalent to running *ahpipeline* with 'instrume=HXI') is run as follows. From the directory above where the original data is located, and for an observation with OBSID ah100050020, issue the command

```
%>hxipipeline indir=100050020 outdir=100050020_repro_hxi
steminputs=ah100050020 stemoutputs=DEFAULT instrument=HXI
entry_stage=1 exit_stage=2 verify_input=no
attitude=100050020/auxil/ah100050020.att.gz
orbit=100050020/auxil/ah100050020.orb.gz hxipipeline
obsgti=../auxil/ah100050020_gen.gti.gz \
makefilter=../auxil/ah100050020.mkf.gz \
extended_housekeeping=../auxil/ah100050020.ehk.gz
```

to apply all of the calibration steps summarized in Table 7.1 to reconstruct the unfiltered event file, and then all of the screening steps summarized in Table 7.4 to reconstruct the cleaned event files. These files are placed in the "repro_hxi" subdirectory indicated by the 'outdir' parameter. If the 'entry_stage' parameter is 2, the data is cleaned but not recalibrated; if the 'exit_stage' parameter is 1 the data is recalibrated but not cleaned. If the 'exit_stage' parameter is 3 the pipeline data products are remade.

7.4 Extracting products

Additional filtering of, and extraction of data products from, cleaned event files may be done using *xselect*. Creating an image should be paired with a cut on energy to reduce the background. In the case of faint, extended objects, the image extracted in the HXI units may be fainter in corners due to the baffle shadow, accounted for in the ARF.

The next step is to define regions and use these to extract source and background spectra. Users should subtract the NXB background, if available, and simultaneously fit the X-ray background and source spectra. Spectra can subsequently be re-binned or grouped using *grppha* or directly within *xselect*.

7.4.1 Preliminaries

The *xselect* session below shows an example of extracting data products from an observation of the compact point source G21.5-0.9. First an image is extracted from a restricted energy band, and source and background regions are made by hand. The background region is used to extract a light curve to look for time variability. Then the background and source spectrum are extracted. This example uses all layers of the HXI. *xselec*t is also used to combine event files from multiple observation sequences of this source prior to data extraction. Extraction of spectra for extended sources may proceed along the same lines, except for the former uses a larger extraction region – or the entire detector.

The barycentric correction may be applied prior to extraction of data products using the *barycen* task, which requires an orbit file and source coordinates as input, as shown in the following example:

```
%>barycen infile=ah100050020hx1_p0camrec_cl.evt.gz
outfile=ah100050020hx1_p0camrec_cl_barycor.evt
orbfile=ah100050020.orb.gz ra=278.38824  dec=-10.570683 orbext=ORBIT
```

Note that the ‘ra’ and ‘dec’ parameter should be set to the true average pointing direction. This should be set to the RA_NOM and DEC_NOM keywords in the header for cases where the attitude is correct, or may be approximated by the object coordinates (RA_OBJ and DEC_OBJ header keywords) in cases where the source is known to be on-axis.

1. **Read in the SXI event list data, examine the 0.5-8 keV image**

```
xsel:SUZAKU > read events g21_sxi_cl.evt
xsel:HITOMI-SXI-WINDOW1 > set xybinsize 4
xsel:HITOMI-SXI-WINDOW1 > filter pha_cutoff 83 1333
xsel:HITOMI-SXI-WINDOW1 > extract image
xsel:HITOMI-SXI-WINDOW1 > plot image
```

The file g21_sxi_cl.evt is the cleaned SXI event file (see previous section).

2. **Visualization: Superimpose the HXI FoV**

Compare the centroid of the source SXI image with the HXI FoV using `ahmkregion`.

```
xsel:HITOMI-SXI-WINDOW1 > $ahmkregion instrume=HXI ra=278.3293 dec=-
10.5790 roll=88.50244
```

where the ra and dec correspond to the centroid of the SXI image (it is assumed that the source is observed approximately on-axis), and the roll angle is obtained from the event file PA_NOM header keyword, e.g.

```
%>fkeyprint g21_sxi_cl.evt PA_NOM
```

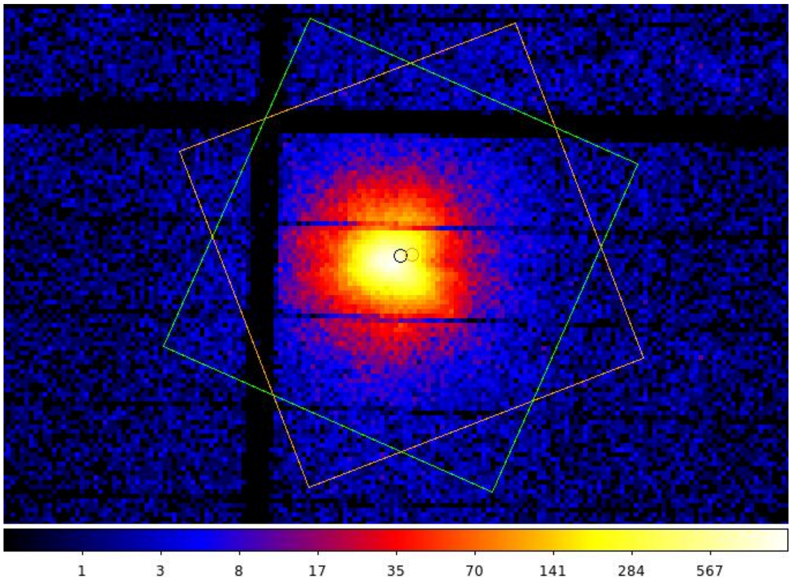


Figure 7.1: G21.5-0.9 SXI image with HXI1 (center: black, border: green) and HXI2 (center and border: orange) FoV superimposed.

3. Select the source and background extraction regions

Extraction regions may selected from examination of the binned 5-80 keV image, either by hand or by calculating the centroid of the image. Example source and background regions are shown below for the HXI2 observation of G21.5-0.9. A circular region of 3 arcmin radius, as recommended, is used for the source spectrum; the remaining portion excluding the edges of the detector for the background. Note that, due the small HXI FoV, the background region will inevitably include some source photons. The same region files may be used for HXI1.

```
xsel:SUZAKU > read events ah100050020hx2_p0camrec_cl.evt.gz
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > set xybinsize 4
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > filter pha_cutoff 50 800
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract image
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > plot image
```

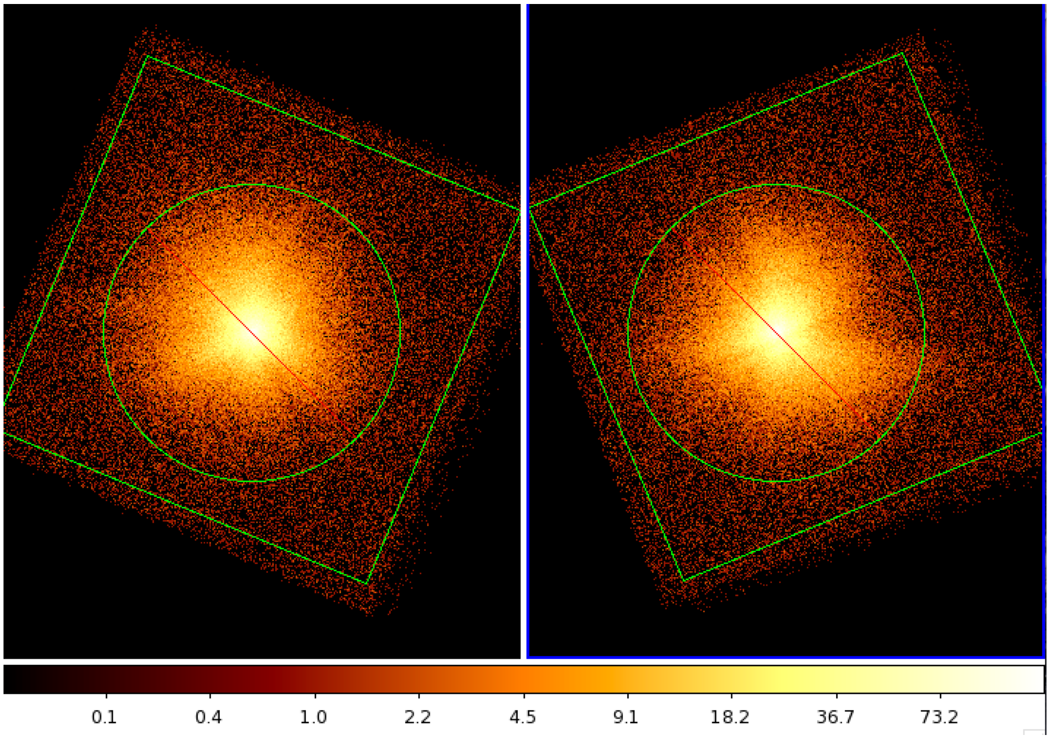


Figure 7.2: G21.5-0.9 5-80 keV HXI2 image with source and background extraction regions shown.

7.4.2 Extract Light Curve and Spectrum

1) Extract an HXI1 source light curve and spectrum for a single sequence using the region file defined above.

```
xsel:SUZAKU > read events ah100050020hx1_p0camrec_cl.evt.gz
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > filter region G21_HXI_source.reg
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract spectrum
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > save spectrum
ah100050020hx1_p0camrec_cl.pi
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > plot spectrum
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract curve
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > save curve
ah100050020hx1_p0camrec_cl.lc
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > plot curve
```

The output light curve and spectrum are show in Figure 7.3.

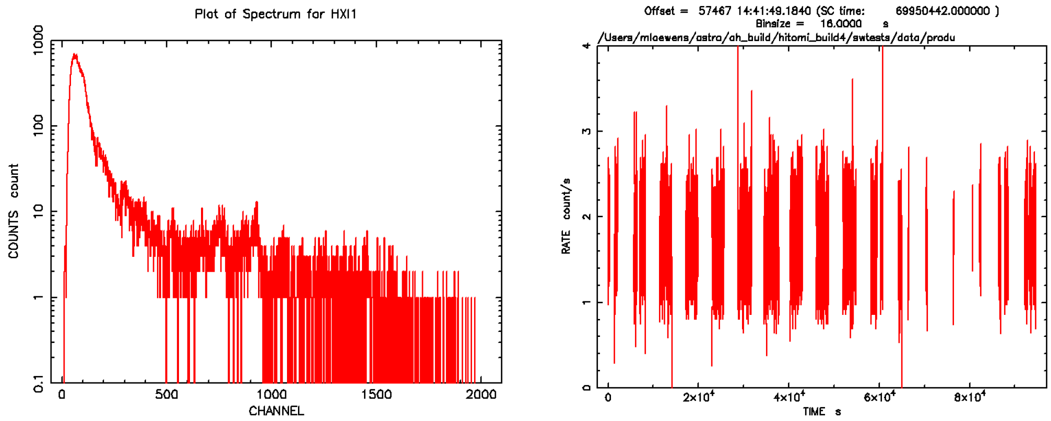


Figure 7.3: HXI1 source spectrum and light curve for G21.5-0.5, sequence 100050020.

2) Extract HXI1 background and source light curves and spectra from event files from multiple sequences using the region file defined above.

```
xsel:SUZAKU > read events ah100050010hx2_p0camrec_cl.evt.gz
xsel:HITOMI-HXI2-CAMERA_NORMAL1 > read events
ah100050020hx2_p0camrec_cl.evt.gz
xsel:HITOMI-HXI2-CAMERA_NORMAL1 > read events
ah100050030hx2_p0camrec_cl.evt.gz
xsel:HITOMI-HXI2-CAMERA_NORMAL1 > filter region G21_HXI_source.reg
xsel:HITOMI-HXI2-CAMERA_NORMAL1 > extract spectrum
xsel:HITOMI-HXI2-CAMERA_NORMAL1 > save spectrum
ah1000500ALL0hx2_p0camrec_cl.pi
xsel:HITOMI-HXI2-CAMERA_NORMAL1 > extract curve
xsel:HITOMI-HXI2-CAMERA_NORMAL1 > save curve
ah1000500ALL0hx2_p0camrec_cl.lc
xsel:HITOMI-HXI2-CAMERA_NORMAL1 > clear region
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > filter region G21_HXI_source.reg
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract spectrum
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > save spectrum
ah1000500ALL0hx2_p0camrec_cl_bkg.pi
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > plot spectrum
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract curve
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > save curve
ah1000500ALL0hx2_p0camrec_cl_bkg.lc
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > plot curve
```

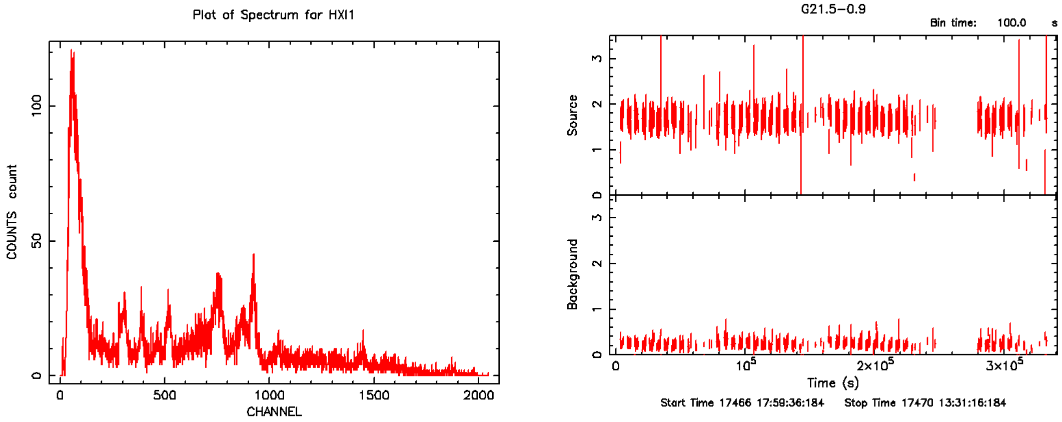


Figure 7.4: HXI1 background spectrum and light curve (compared to source light curve) for G21.5-0.5, combined sequences 100050010, 100050020, and 100050030.

7.4.3 Additional filtering

The event files can be further screened by layer. In the example below the G21.5-0.9 HXI1 data products are extracted, first, for events in the Si layers 0:3 and, second, in the CdTe layer 4:4. Note that there is no current capability for creating response files for a selection of layers.

```
xsel:SUZAKU > read events ah100050020hx1_p0camrec_cl.evt.gz
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > filter column "LAYER=0:3"
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > filter region G21_HXI_source.reg
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract curve
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > save  curve
ah100050020hx1_p0camrec_cl_Si.lc
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract spectrum
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > save spectrum
ah100050020hx1_p0camrec_cl_Si.pi
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > clear column
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > filter column "LAYER=4:4"
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract curve
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > save  curve
ah100050020hx1_p0camrec_cl_CdTe.lc
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > extract spectrum
xsel:HITOMI-HXI1-CAMERA_NORMAL1 > save spectrum
ah100050020hx1_p0camrec_cl_CdTe.pi
```

7.4.4 Dead Time and Barycenter Corrections

The task *hxisgddtime* applies a dead time correction to HXI (and SGD) light curves and spectra. Inputs to this task include a cleaned pseudo-event file, an extracted light curve and/or spectrum, and a file with a GTI extension or a text file containing a list of GTI. The output includes the corrected light curve and a spectrum. This deadtime correction is not applicable for short time bin (<16 sec, parameter) lightcurves. An example, applied to a single G21.5-0.5 sequence (100050020) for the HXI1 detector, is as follows.

```
%> hxisgddtime infile=ah100050020hx1_p0camrecpse_cl.evt.gz \
inlcfile=ah100050020hx1_p0camrec_cl.lc \
inspecfile=ah100050020hx1_p0camrec_cl.pi \
outlcfile=ah100050020hx1_p0camrec_dtime.lc \
outfile=ah100050020hx1_p0camrec_dtime.pi \
gtifile=ah100050020hx1_p0camrec_cl.evt.gz
```

If not already applied to the event file, the barycenter correction may be applied to the dead-time corrected light curve using barycen.

```
%>barycen infile=ah100050020hx1_p0camrec_dtime.lc \
outfile=ah100050020hx1_p0camrec_dtime_add_bary.lc \
orbfile=ah100050020.orb.gz ra=278.38824 dec=-10.570683 orbext=ORBIT
```

When background and source light curves and spectra are extracted from event files from multiple sequences as in Section 7.4.2 example 2, a two-step process is required to correctly apply the multiple pseudo-event files. The following example does this for the same three sequences, again for the case of HXI1. Note that both source and background files are corrected.

First, merge the pseudo-event files, including their GTI extensions.

```
%>ftmerge ah100050010hx1_p0camrecpse_cl.evt.gz, \
ah100050020hx1_p0camrecpse_cl.evt.gz, \
ah100050030hx1_p0camrecpse_cl.evt.gz \
ah1000500ALL0hx1_p0camrecpse_cl.evt
```

```
%>ahgtigen infile=NONE outfile=ah1000500ALL0hx1_p0camrecpse_cl.gti \
gtifile=@ah10005001230hx1_p0camrecpse_cl.gti.lst gtiexpr=NONE \
mergegti=OR
```

where ah10005001230hx1_p0camrecpse_cl.gti.lst is the following text file listing all pseudo-event file GTI extensions:

```
ah100050010hx1_p0camrecpse_cl.evt.gz+2
ah100050020hx1_p0camrecpse_cl.evt.gz+2
ah100050030hx1_p0camrecpse_cl.evt.gz+2
```

```
%>ftdelhdu 'ah1000500ALL0hx1_p0camrecpse_cl.evt[GTI]' none \
confirm=YES
```

```
%>ftappend 'ah1000500ALL0hx1_p0camrecpse_cl.gti[GTI]' \
ah1000500123450hx1_p0camrecpse_cl.evt
```

Next, merge the event file GTI extensions.

```
%>ahgtigen infile=NONE outfile=ah1000500ALL0hx1_p0camrec_cl.gti
gtifile=@ah10005001230hx1_p0camrec_cl.gti.lst gtiexpr=NONE mergegti=OR
```

where `ah10005001230hx1_p0camrec_cl.gti.lst` is a text file listing all GTI extensions:

ah100050010hx1_p0camrec_cl.evt.gz+2
ah100050020hx1_p0camrec_cl.evt.gz+2
ah100050030hx1_p0camrec_cl.evt.gz+2

```
fthedit ah1000500ALL0hx1_p0camrec_cl.gti+1 INSTRUME a HXI1
fthedit ah1000500ALL0hx1_p0camrec_cl.gti+1 DETNAM a CAMERA
```

The deadtime correction may now be applied to the merged source and background spectra and lightcurves, using the merged pseudo-event files, as follows:

```
%>hxisgddtime \
infile=ah1000500ALL0hx1_p0camrecpse_cl.evt \
inlcfile=ah1000500ALL0hx1_p0camrec_cl.lc \
inspecfile=ah1000500ALL0hx1_p0camrec_cl.pi \
outlcfile=ah1000500ALL0hx1_p0camrec_dtime.lc \
outfile=ah1000500ALL0hx1_p0camrec_dtime.pi \
gtifile=ah1000500ALL0hx1_p0camrec_cl.gti

%>hxisgddtime \
infile=ah1000500ALL0hx1_p0camrecpse_cl.evt \
inlcfile=ah1000500ALL0hx1_p0camrec_cl_bkg.lc \
inspecfile=ah1000500ALL0hx1_p0camrec_cl_bkg.pi \
outlcfile=ah1000500ALL0hx1_p0camrec_dtime_bkg.lc \
outfile=ah1000500ALL0hx1_p0camrec_dtime_bkg.pi \
gtifile=ah1000500ALL0hx1_p0camrec_cl.gti
```

7.5 Background Estimation

Both external astrophysical and internal non-X-ray (NXB) components contribute to the total background associated with any observation. The local background, including the NXB, may be estimated using a spectrum extracted identically to the source region but from a separate off-source region as in the example is Section 7.4. However due to the small HXI field-of-view such a background includes some source photons – even for a point source – and thus overestimates the background. As a result, HXI spectral analysis should include subtracting the NXB background and simultaneously fitting the X-ray background and source spectra, i.e. modeling the X-ray background. The X-ray background should, at the least, include a power-law with a high-energy cutoff representing the extragalactic background component.

The NXB spectrum may be estimated using the task *hxinxbgen*. This task requires the following input files: (1) an event file for which the NXB is to be calculated (used to provide header keywords related to the time of the observation, and the pointing, as well as the GTI), (2) an EHK file from the auxil directory, (3) a region file in either SKY or FOC coordinates (matching the value of the 'regmode' parameter, (4) an NXB events file (containing events derived from the pseudo-trigger and found in the trend data) and (5) an NXB EHK file, also from the trend data. The NXB filtering must match that used to derive the source spectrum. *hxinxbgen* outputs the weighted NXB PI spectrum; and optionally, the EHK file corresponding to the science GTI, the calibrated, screened, and time-filtered NXB event list ('outnxbfile'); and the NXB EHK file corresponding to that NXB file ('outnxbehk').

Returning to the first example in Section 7.4, the command to create the NXB spectrum is:

```
%>hxinxbgen infile=ah100050020hx1_p0camrec_cl.evt.gz \
ehkfile=ah100050020.ehk.gz regfile=G21_HXI_source.reg \
innxbfile=nxb.fits.gz innxbehk=nxb.ehk.gz inpsefile=nxbpse.evt.gz \
outpifile=ah100050020hx1_p0camrec_cl_nxb.pi \
sortcol="COR2" sortbin="0,4,5,6,7,8,9,10,11,12,13,99"
```

The 'sortcol' and 'sortbin' parameters are chosen to match the COR2 filtering.

Note that, because of the short HITOMI lifetime, the hxinxbgen tool is not relevant. Standard HXI background spectra based on observations with no source may be available in the future.

7.6 RSP generation

For the HXI units the net response matrix (RSP) file, rather than separate RMF and ARF files, is created using the tool *aharfgen*. The following example illustrates the commands required to make an RSP file for an HXI (in this case, HXI1) spectrum. Some of the hidden parameters are also shown for completeness.

The first step is to run *ahexpmap*, as in the following example. The input parameters include the extended HK file for this observation (parameter 'ehkfile'), the cleaned event file that provides the GTI (parameter 'gtifile')), and the attitude binning parameters 'delta and 'numphi'.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz \
gtifile=ah100050020hx1_p0camrec_cl.evt.gz instrume=HXI1 \
badimgfile=NONE pixgtifile=NONE outfile=ah100050020hx1_p0camrec.expo \
outmaptype=EXPOSURE delta=20.0 numphi=1 stopsys=SKY instmap=CALDB \
qefile=CALDB contamifile=CALDB vigfile=CALDB obffile=CALDB \
fwfile=CALDB gvfile=CALDB maskcalsrc=yes fwtype=DEFAULT \
specmode=MONO specfile=spec.fits specform=FITS energy=10.0 \
evperchan=DEFAULT abund=1 cols=0 covfac=1
```

If the attitude is stable, or inaccurate based on a comparison of the centroid of the SXI image with the source coordinates, one can run *ahexpmap* with 'numphi=1' and a large value of 'delta' (~100) and so that there is a single attitude bin in the output histogram. Generally the 'FRACTION' column in the 'OFFAXISHIST' extension of the exposure map should be examined, and the value of 'delta' adjusted to eliminate entries with values <<1 to save computing time when constructing the ARF.

The next step is to run *aharfgen,* a script that internally calls *hxirspeffimg* to create the RSP file ah100050020hx1_point.rsp. The following example produces an RSP for a point source (parameter 'sourcetype=POINT'). The input parameters include the source coordinates (parameters 'source_ra' and 'source_dec'), the delta-attitude and CAMS-offset files (parameters 'filtoffsetfile' and 'dattfile'), the exposure map produced in the previous step (parameter 'emapfile'), the extraction region in SKY coordinates (parameter 'regionfile'), and the energy range (parameter 'erange'). If the attitude is suspect, the centroid of the source image should be used in place of the true source coordinates. The expected runtime for the set of parameters below is ~30-40 minutes per attitude bin, and the value of 'numphoton' should be adjusted accordingly.

```
%>aharfgen xrtevtfile=raytrace_ah100050020hx1_ptsrc_evt.fits \
source_ra=278.38824  source_dec=-10.570683  telescop=HITOMI \
instrume=HXI1 emapfile=ah100050020hx1_p0camrec.expo \
dattfile=ah100050020hx1.att.gz regmode=SKY \
regionfile=region_HXI_100050020_sky.reg sampling=120 \
sourcetype=point erange="4.0 80.0 0 0" \
outfile=ah100050020hx1_point.rsp \
filtoffsetfile=ah100050020hx1_cms.fits.gz numphoton=10000 \
minphoton=1 teldeffile=CALDB qefile=CALDB rmffile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB
```

The following example produces an RSP for a uniform circular distribution of 10 arcminute radius ('sourcetype=POINT', 'flatradius=10.0'), ah100050020hx1_p0camrec_flat.arf.

```
%>aharfgen xrtevtfile=raytrace_ah100050020hx1_flat_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=HXI1 emapfile=ah100050020hx1_p0camrec.expo \
dattfile=ah100050020hx1.att.gz regmode=SKY \
regionfile=region_HXI_100050020_sky.reg sampling=120 \
sourcetype=FLATCIRCLE  flatradius=10.0 erange="4.0 80.0 0 0" \
outfile=ah100050020hx1_flat.rsp \
filtoffsetfile=ah100050020hx1_cms.fits.gz numphoton=10000 \
minphoton=1 teldeffile=CALDB qefile=CALDB rmffile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB
```

The following example produces an RSP for a beta-model distribution of core radius 1 arcminute radius, beta parameter 0.6, and maximum radius 10 arcminutes ('sourcetype=BETAMODEL, 'betapars="1.0,0.6,10.0" '), ah100050020hx1_p0camrec_beta.rsp.

```
%>aharfgen xrtevtfile=raytrace_ah100050020hx1_beta_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=HXI1 emapfile=ah100050020hx1_p0camrec.expo \
dattfile=ah100050020hx1.att.gz regmode=SKY \
regionfile=region_HXI_100050020_sky.reg sampling=120 \
sourcetype=BETAMODEL \
betapars=”1.0,0.6,10.0” erange="4.0 80.0 0 0" \
outfile=ah100050020hxl_beta.rsp \
filtoffsetfile=ah100050020hxl_cms.fits.gz numphoton=10000 \
minphoton=1 teldeffile=CALDB qefile=CALDB rmffile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB
```

The following example produces an ARF for an input 2-8 keV image, image.fits arcminutes (‘sourcetype=image’, ‘imgfile=image.fits’), ah100050020hx1_p0camrec_image.rsp. The third and fourth entries in the erange parameter correspond to the image bandpass (here, 2-8 keV).

```
%>aharfgen xrtevtfile=raytrace_ah100050020hxl_image_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=HXI1 emapfile=ah100050020hxl_p0camrec.expo \
dattfile=ah100050020hxl.att.gz regmode=SKY \
regionfile=region_HXI_100050020_sky.reg sampling=120 \
sourcetype=image imgfile=image.fits erange="4.0 80.0 2.0 8.0" \
outfile=ah100050020hxl_p0camrec_image.rsp \
filtoffsetfile=ah100050020hxl_cms.fits.gz numphoton=10000 \
minphoton=1 teldeffile=CALDB qefile=CALDB rmffile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB
```

7.7 Summary of tasks used in this chapter

Below is the list of all the tasks used or mentioned in this chapter. Readers are referred to the help files of these tasks for a more extensive description.

- aharfgen: Make an ancillary response function (ARF) file for the SXS or SXI, or a response matrix (RSP) file for the HXI
- ahexpmap: Generate an exposure map for HXI, SXI, and SXS, or a flat field image for SXI and SXS
- ahgtigen: Create and/or merge GTI files
- ahmkregion: Create regions files showing the field of view of all Hitomi instruments
- ahpipeline: HXI/SGD/SXS/SXI reprocessing tool
- barycen: Correct time to Solar System barycenter
- cams2att: Compute a time-dependent delta-attitude file
- coordevt: Convert events from one coordinate system to another
- fkeyprint: Print the specified keyword(s) in the headers of a list of input FITS files
- ftappend: Append a copy of an HDU from one file onto another file
- ftcopy: Copy the contents of a FITS file to a new file
- ftdelhdu: Delete a HDU (header-data unit) in a FITS file
- ftmerge: Merge (append) rows from multiple input tables into a single output table

- grppha: Manipulate OGIP standard PHA FITS file
- hxievtid; Reconstruct HXI events.
- hxinxbgen: Create a Non-X-ray Background (NXB) spectrum for HXI
- hxipipeline: HXI reprocessing tool
- hxirspeffimg: Create a response and/or flat-field correction image fro the HXI
- hxisgddtime: Calculate and correct for deadtime HXI and SGD spectra and lightcurves
- hxisgdpha: Calibrate the HXI or SGD PHA for each signal in the SFF event file
- hxisgdsff: Convert an HXI or SGD First FITS File (FFF) into the Second FITS File (SFF)
- xselect: Extract image or spectrum for further analysis

8 SOFT GAMMA-RAY DETECTOR (SGD)

8.1 Introduction

The Soft Gamma-ray Detector (SGD) operates in the energy range from 40-600 keV. There are two separate SGD units, one on each side of the satellite. Each SGD unit consists of three Compton Cameras surrounded by an active Shield. The central detector of the SGD is the Compton Camera (CC), which is composed of layered Si and CdTe pixelated sensors with <2 keV energy resolution. A fine collimator restricts the SGD field of view to 66 X 66 arcmin at 40 keV, with some collimation up to 150 keV. The hybrid design of the Compton Camera module incorporates both pixelated Si and CdTe detectors. The Si sensors are used as scatterers since Compton scattering dominates over photoelectric absorption in Si above ~50 keV. The CdTe sensors are used as absorbers of the gamma-ray photon following the Compton scattering in the Si sensors. Although the SGD is not an imaging detector, two-dimensional spatial sensitivity is necessary for event reconstruction. Therefore the Si and CdTe sensors are pixelated with a pixel size of 3.2×3.2 mm^2 for both sensors, and thickness 0.6 mm for Si and 0.75 mm for CdTe. Each incoming or secondary gamma-ray above threshold that impacts one of the sensors activates the semiconductor material and liberates a charge, which is collected by an implanted cathode and read out using an application-specific integrated circuit (ASIC). Depending on the physics of the initial interaction, one or more sensor layers is impacted as the result of a single incident photon. The event is reconstructed on the ground to determine the energy of the initial gamma-ray and to distinguish between gamma-ray-induced hit patterns and hit patterns arising from internal background events. The Active Shield, which consists of Bismuth Germanium Oxide (BGO) and fully encloses the Compton Cameras, except for the front aperture, further reduces the background by rejecting the majority of external background events. Additional reduction is achieved below 150 keV since the front aperture to the CC is constrained by a passive fine collimator

8.2 Cleaned Event File Content

The files referred to in this chapter have been calibrated and filtered by the standard screening procedure. The primary Hitomi tools used in the calibration steps are summarized in Table 8.1. These were applied to the first files that were converted from Hitomi telemetry, and had time assigned and various keywords set. The resulting files are placed in the sgd/event_uf subdirectory. After unpacking the telemetry using *hxisgdsff* and assigning energy with *hxisgdpha*, each occurrence in the telemetered event file is reconstructed into a single photon interaction event using the tool *sgdevtid*. This tool determines the signal validity of an occurrence and computes the energy of a photon detected in the SGD and the 3-dimensional coordinates of the first photon interaction. In the cleaned event file, each row corresponds to a reconstructed event and the columns 'CAMERAX', 'CAMERAY', 'CAMERAZ', and 'PI' contain the reconstructed position and energy of that event along with additional columns containing information related to the number and the type of interactions. If the algorithm is unable to reconstruct a valid event, then the ‘PI’ column value is set to NULL and diagnostic information is written to the output file. These NULL ‘PI’ events are filtered out in the event cleaning.

Calibration Step	Tool	Comments
Unpack telemetry	`hxisgdsff`	Output data stored in variable-length arrays
Compute EPI	`hxisgdpha`	Also flags bad pixels
Reconstruct events	`sgdevtid`	Derives position and total deposited energy for events that can be properly reconstructed.

Table 8.1: Primary calibration steps.

The columns that can be used for data filtering are listed in Table 8.2.

Column	Description
`DELCOMPTON`	Error on difference between geometric and kinematic scattering angles
`COMPTON_TH`	Kinematic Compton scattering angle (θ); range 0° < θ < 180°
`COMPTON_PH`	Kinematic Compton scattering angle (ϕ); range -180° < ϕ < 180°
`OFFAXIS`	Angle (degrees) between line of sight and Compton scattering angle
`LIKELIHOOD`	Likelihood of event (0:1)
`MATTYPE`	Material where the event originated: 1 =Si Layer, 2 =CdTe layer, 3 = multiple layers.

Table 8.2: Columns used for additional filtering.

The cleaned event files, located in the sgd/event_cl subdirectory, are separated by SGD unit and by Compton Camera. There are also two files for each CC, normal (photon) events and pseudo events, e.g. ah100050020sg1_p0cc1rec_cl.evt.gz and ah000500020sg1_p0cc1recpse_cl.evt.gz contain reconstructed normal and pseudo-events, respectively, for SGD unit 1, Compton Camera 1. Only the photon events are to be used for science analysis; the pseudo-events are used for dead time correction. Both normal and pseudo-event files have the same format and the same columns.

The reconstructed unfiltered event files (e.g. ah100050020sg1_p0cc1rec_ufa.evt) in the sgd/event_uf directory are screened using the criteria given in Table 8.3 (for SGD-[n], CC-[m]) to produce the cleaned event files using two broad classes of screening: event-by-event and by good-time intervals (GTI). The GTI screening filters slew and bad attitude intervals, and intervals of saturated telemetry. The GTI files needed for this screening are already present in the auxil directory or sgd/event_uf directories. In addition, GTI corresponding to nominal instrument status based on housekeeping data encapsulated in the makefilter (mkf) file in the auxil directory, and GTI-based on orbit- and pointing-derived criteria (times of stable pointing, unblocked by the earth, outside of regions of high background) encapsulated in the extended housekeeping (ehk) file in the auxil directory are also applied. The GTI constructed from ehk and mkf files are remade in the course of re-screening the data – either automatically (if using the *ahpipeline* script), or manually.

Type	File	Criterion	Comments

event-based	event_uf/OBSIDsg[n]_cc[m]uf.evt	FLAG_LCHKMIO == b0 && FLAG_CCBUSY == b0 && FLAG_HITPAT_CC[1] == 1 && FLAG_HITPAT_CC[2] == 0 && FLAG_HITPAT_CC[3] == 0 && FLAG_HITPAT == b0 && FLAG_FASTBGO == b0 && FLAG_SEU == b0 && FLAG_LCHK == b0 && FLAG_CALMODE == b0 && FLAG_TRIGPAT[29] == 0 && FLAG_TRIGPAT[30] == 0 && FLAG_TRIGPAT[31] == 0 && LIKELIHOOD <= 1.0 && PROC_STATUS[1] = 0 && PROC_STATUS[2] = 0 && STATUS[3]==b0 && NUMHITS[1]==b0	MIO Length check no error; CC not busy with data processing; HitPattern_CC = 1 for CCn and HitPattern_CC = 0 for the other two CCs (m ≠ n); No shield signals; No fast BGO signals; No single event upset; Length check no error; Normal operation mode; No forced trigger; No pseudo trigger; No calibration pulse; Likelihood <= 1.0; Proc_status bits not set; Status bits not set; Number of hits = 1
GTI-based	auxil/OBSID.mkf	SGDn_USR_DE_MODE == 1 && SGDn_USR_CCm_MODE == 2 && SGDn_USR_CPMU_HVp_CAL >= 200.0 && SGDn_USR_APMU1_HVq_CAL >= 200.0 && SGDn_USR_APMU2_HVq_CAL >= 200.0	Observing mode for SGDn Ddtector and CC; All SGDn CPMU, APMU1 & APMU2 HV >= 200.0 V (SGDn=1,2; CCm=1,2,3; HVp= 1,2,3,4; HVq=1,2)
GTI-based	auxil/OBSID.ehk	ANG_DIST < 1.5 && SAA_SGDn==0 && T_SAA_SGDn>251 && TN_SAA_SGDn > 251	Less than 1.5 arcminutes from mean pointing position; Outside of SAA and more than 251 seconds from SAA passage;

		&& ELV>5	More than 5 degrees Earth elevation
GTI-based	auxil/OBSID_gen.gti	inside GTIPOINT and GTIATT	Excludes slew and bad attitude intervals
GTI-based	auxil/OBSIDsgd_tel.gti	inside GTITEL	Excludes times of telemetry saturation

Table 8.2: Screening criteria. : Screening criteria for SGD unit 'n', Compton Camera 'm'.

The screening is done separately for each SGD unit (n=1,2) and each CC within an SGD unit (m=1,2,3). The screening above is that applied to the normal events.

Reprocessing of the unfiltered events files is necessary if updates to relevant SGD CALDB files were made since the cleaned files were produced (see explanation of CALDBVER in Section 3.2) by checking http://heasarc.gsfc.nasa.gov/docs/hitomi/calib/ and the list of calibration files used in the SGD pipeline shown in Table 8.3.

8.3 Reprocessing Events

The calibration files used in SGD pipeline processing are listed in Table 8.3.

Tool	Calibration File	Description
hxisgdsff	ah_sgd_remap_yyyymmddv00n.fits	SGD readout remapping file
hxisgdpha	ah_sgn_gain_yyyymmddv00n.fits ah_sgn_badpix_yyyymmddv00n.fits	SGD PHA calibration gain function Flags for active/dead/noisy readout channels
sgdevtid	ah_sgd_remap_yyyymmddv00n.fits ah_sgd_line_yyyymmddv00n.fits ah_sgn_badpix_yyyymmddv00n.fits ah_sgd_probseq_yyyymmddv002.fits ah_sgd_probfov_YYYYMMDDv001.fits	SGD readout remapping file SGD Fluorescence file SGD energy threshold file SGD sequence probability file SGD FoV probability file

Table 8.3: Calibration database files used in primary calibration steps for SGD1 (n=1) and SGD2 (n=2).

The most straightforward approach is using the *sgdpipeline* script (equivalent to running *ahpipeline* with instrume=SGD). From the directory above where the original data is located, and for an observation with OBSID ah100050020, issue the command

```
%>sgdpipeline indir=100050020 outdir=100050020_repro_sgd \
steminputs=ah100050020 stemoutputs=DEFAULT instrument=SGD \
entry_stage=1 exit_stage=2 verify_input=no
\obsgti=../auxil/ah100050020_gen.gti.gz \
makefilter=../auxil/ah100050020.mkf.gz \
extended_housekeeping=../auxil/ah100050020.ehk.gz
```

to apply all of the calibration steps summarized in Table 8.1 to reconstruct the unfiltered event file, and then all of the screening steps summarized in Table 8.2 to reconstruct the cleaned event files. These files are placed in the “repro_sgd” subdirectory indicated by the ‘outdir’ parameter. If the ‘entry_stage’ parameter is 2, the data is cleaned but not recalibrated; if the ‘exit_stage’ parameter is 1, the data is recalibrated but not cleaned. If the ‘exit_stage’ parameter is 3, the pipeline data products are remade.

8.4 Extracting products

The next step is to extract the source spectrum. Users should subtract the NXB background, if available, and simultaneously fit the X-ray background and source spectra. Spectra can subsequently be re-binned or grouped using *grppha* or directly within *xselect*.
SGD event files are separated by SGD unit (1 or 2) and by Compton Camera (CC; 1, 2, or 3). The analysis may be done on each CC separately and the resulting spectra fit simultaneously, or they may be co-added.

The barycentric correction may be applied prior to extraction of data products using the *barycen* task, which requires an orbit file and source coordinates as input, as shown in the following example:

```
%>barycen infile=ah100050020sg1_p0cc2rec_cl.evt.gz
outfile=ah100050020sg1_p0cc2rec_cl_barycor.evt
orbfile=ah100050020.orb.gz ra=278.38824  dec=-10.570683 orbext=ORBIT
```

Note that the ‘ra’ and ‘dec’ parameters should be set to the true average pointing direction. This should be set to the RA_NOM and DEC_NOM keywords in the header for cases where the attitude is correct, or may be approximated by the object in cases where the source is known to be on-axis.

8.4.1 Extract Light Curve and Spectrum

The *xselect* session below shows an example of extracting spectra and light curves from an observation of the compact point source G21.5-0.9. In the following, *xselect* is used to extract light curves and spectra from individual Compton Cameras, and also to combine event files from multiple CC prior to data extraction.

1) Extract separate products for the event file for each SGD1 Compton Camera.

```
xsel:SUZAKU > read events ah100050020sg1_p0cc1rec_cl.evt.gz
xsel:HITOMI-SGD1-CC_NORMAL1 > extract spectrum
xsel:HITOMI-SGD1-CC_NORMAL1 > save spectrum \
ah100050020sg1_p0cc1rec_cl.pi
xsel:HITOMI-SGD1-CC_NORMAL1 > extract curve
xsel:HITOMI-SGD1-CC_NORMAL1 > save curve ah100050020sg1_p0cc1rec_cl.lc
xsel:HITOMI-SGD1-CC_NORMAL1 > clear all
xsel:HITOMI-SGD1-CC_NORMAL1 > read events \
ah100050020sg1_p0cc3rec_cl.evt.gz
xsel:HITOMI-SGD1-CC_NORMAL1 > extract spectrum
```

```
xsel:HITOMI-SGD1-CC_NORMAL1 > save spectrum
\ah100050020sg1_p0cc2rec_cl.pi
xsel:HITOMI-SGD1-CC_NORMAL1 > extract curve
xsel:HITOMI-SGD1-CC_NORMAL1 > save curve ah100050020sg1_p0cc2rec_cl.lc
xsel:HITOMI-SGD1-CC_NORMAL1 > clear all
xsel:HITOMI-SGD1-CC_NORMAL1 > read events \
ah100050020sg1_p0cc3rec_cl.evt.gz
xsel:HITOMI-SGD1-CC_NORMAL1 > extract spectrum
xsel:HITOMI-SGD1-CC_NORMAL1 > save spectrum \
ah100050020sg1_p0cc3rec_cl.pi
xsel:HITOMI-SGD1-CC_NORMAL1 > extract curve
xsel:HITOMI-SGD1-CC_NORMAL1 > save curve ah100050020sg1_p0cc3rec_cl.lc
```

2) Extract products from the SGD1 combined event file.

```
xsel:SUZAKU > read events ah100050020sg1_p0cc1rec_cl.evt.gz
xsel:HITOMI-SGD1-CC_NORMAL1 > read events \
ah100050020sg1_p0cc2rec_cl.evt
xsel:HITOMI-SGD1-CC_NORMAL1 > read events \
ah100050020sg1_p0cc3rec_cl.evt
xsel:HITOMI-SGD1-CC_NORMAL1 > save events \
ah100050020sg1_p0ccALLrec_cl.evt
xsel:HITOMI-SGD1-CC_NORMAL1 > extract curve
xsel:HITOMI-SGD1-CC_NORMAL1 > plot curve
xsel:HITOMI-SGD1-CC_NORMAL1 > save curve
\ah100050020sg1_p0ccALLrec_cl.lc
xsel:HITOMI-SGD1-CC_NORMAL1 > extract spectrum
xsel:HITOMI-SGD1-CC_NORMAL1 > save spectrum \
ah100050020sg1_p0cc3rec_cl.pi
```

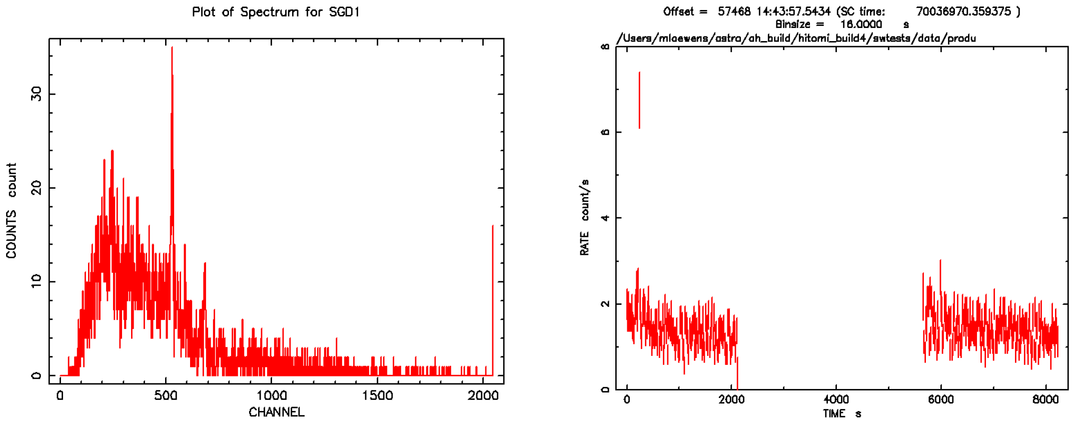


Figure 8.5: SGD1 combined camera source spectrum and lightcurve for G21.5-0.5, sequence 100050020.

8.4.2 Dead Time Correction

The task *hxisgddtime* applies a dead time correction to SGD (and HXI) lightcurves and spectra. Inputs to this task include a cleaned pseudo-event file, an extracted light curve and spectrum, and a file with a GTI extension or a text file containing a list of GTI. The output includes the corrected light curve and spectrum. This deadtime correction is not applicable for short time bin (<16 sec, mintimedel parameter) light curves. An example, applied to a single G21.5-0.5 sequence (100050020) for the SGD1 detector, is as follows.

Apply the deadtime correction to the lightcurve and spectrum for each Compton Camera.

```
%> hxisgddtime infile=ah100050020sg1_p0cc1recpse_cl.evt
inlcfile=ah100050020sg1_p0cc1rec_cl.lc
inspecfile=ah100050020sg1_p0cc1rec_cl.pi
outlcfile=ah100050020sg1_p0cc1rec_dtime.lc
outfile=ah100050020sg1_p0cc1rec_dtime.pi
gtifile=ah100050020sg1_p0cc1rec_cl.evt chatter=2 clobber=yes
```

```
%> hxisgddtime infile=ah100050020sg1_p0cc2recpse_cl.evt
inlcfile=ah100050020sg1_p0cc2rec_cl.lc
inspecfile=ah100050020sg1_p0cc2rec_cl.pi
outlcfile=ah100050020sg1_p0cc2_dtime.lc
outfile=ah100050020sg1_p0cc2_dtime.pi
gtifile=ah100050020sg1_p0cc2_cl.evt chatter=2 clobber=yes
```

```
%> hxisgddtime infile=ah100050020sg1_p0cc3recpse_cl.evt
inlcfile=ah100050020sg1_p0cc3rec_cl.lc
inspecfile=ah100050020sg1_p0cc3rec_cl.pi
outlcfile=ah100050020sg1_p0cc3_dtime.lc
outfile=ah100050020sg1_p0cc3_dtime.pi
gtifile=ah100050020sg1_p0cc3_cl.evt chatter=2 clobber=yes
```

Add the individual spectra using `mathpha` with exposure set to the average of the three individual spectra.

```
%>mathpha
expr=ah100050020sg1_p0cc1rec_dtime.pi+ah100050020sg1_p0cc1rec_dtime.pi
+ah100050020sg1_p0cc1rec_dtime.pi units=C
outfil=ah100050020sg1_p0ccALL_dtime.pi exposure=2962.33  areascal=%
backscal=% ncomments=0
```

Keywords required by the SGD ARF generator (*sgdarfgen*) are read from the header of any of the individual spectra as identified using *fkeyprint,* e.g.

```
%>fkeyprint ah100050020sg1_p0cc2rec_dtime.pi PA_NOM
```

and added to the cumulative spectrum.

```
%>fthedit ah100050020sg1_p0ccALLrec_dtime.pi+1 RA_NOM a
278.384889916054
%>fthedit ah100050020sg1_p0ccALLrec_dtime.pi+1 DEC_NOM a -
10.5700436450391
%>fthedit ah100050020sg1_p0ccALLrec_dtime.pi+1 PA_NOM a
88.4919021035179
%>fthedit ah100050020sg1_p0ccALLrec_dtime.pi+1 DATE-OBS a 2016-03-
21T14:42:41.359375
```

If not already applied to the event file, the barycenter correction may be applied to the dead-time corrected light curve using `barycen`.

```
%>barycen infile=ah100050020sg1_p0cc3_dtime.lc \
outfile=ah100050020sg1_p0cc3_dtime_add_bary.lc \
orbfile=ah100050020.orb.gz ra=278.38824  dec=-10.570683 orbext=ORBIT
```

8.5 Background estimation

Both external astrophysical and internal non-X-ray (NXB) components contribute to the total background associated with any observation. SGD spectral analysis should include subtracting the NXB background and simultaneously fitting the X-ray background and source spectra, i.e. modeling the X-ray background. The X-ray background should include a power-law with a high-energy cutoff representing the extragalactic background component.

8.6 RSP generation

When the source of interest is on-axis, the pre-computed response matrices stored in CALDB (under the data/hitomi/sgd/cpf/response subdirectory) should be used. These matrices are available either as six separate response files (one per SGD unit per CC), or as a single merged response suitable for analysis of a merged spectrum from all six CCs.

However, if the source is off-axis, then the user should compute an instrument response using the task *sgdarfgen*, which can calculate either a response for a single CC or three separate responses for three CCs. If a merged response file is required, the user must merge the three output response files. The input to *sgdarfgen* includes an input file (e.g., an event file or PI spectrum) with the nominal pointing RA, DEC, and roll angle; a single on-axis RSP file or a list, the source RA and DEC, the SGD unit (1 or 2), and CC id (1, 2, or 3 – or 0 to compute for all 3). The command

```
%>sgdarfgen infile=ah100050020sg1_p0ccALLrec_cl.pi
rspfile="$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc1_20140101v001.r
sp,$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc2_20140101v001.rsp,$CA
LDB/data/hitomi/sgd/cpf/response/ah_sg1_cc3_20140101v001.rsp"
outfile=outrsp ra=278.38824  dec=-10.570683 sgdid=1 ccid=0
```

creates the output response files outrsp_sgd1_cc1.rsp, outrsp_sgd1_cc2.rsp, and outrsp_sgd1_cc3.rsp, which may be combined for application to the merged SGD1 spectrum, as calculated above, via the command

```
%>addrmf outrsp_sgd1_cc1.rsp,outrsp_sgd1_cc2.rsp,outrsp_sgd1_cc3.rsp
1.0,1.0,1.0 ah_sg1_ccALL_20140101v001.rsp
```

8.7 Summary of tasks used in this chapter

Below is the list of all the tasks used or mentioned in this chapter. Readers are referred to the help files of these tasks for a more extensive description.

- addrmf: Sum RMF or RSP files with specific weights
- ahmkregion: Create regions files showing the field of view of all Hitomi instruments
- ahpipeline: HXI/SGD/SXS/SXI reprocessing tool
- barycen: Correct time to Solar System barycenter
- ftcopy: Copy the contents of a FITS file to a new file
- grppha: Manipulate OGIP standard PHA FITS file
- hxisgddtime: Calculate and correct for deadtime HXI and SGD spectra and lightcurve
- hxisgdpha: Calibrate the HXI or SGD PHA for each signal in the SFF event file
- hxisgdsff: Convert an HXI or SGD First FITS File (FFF) into the Second FITS File (SFF)
- mathpha: Perform mathematical operations on PHA files
- sgdarfgen: Generate SGD response by correcting the onaxis response according to the transmission ratio for a given source
- sgdpipeline: SGD reprocessing tool
- xselect: Extract image or spectrum for further analysis

9 Summary of Analysis steps

This chapter describes the minimum steps for analyzing point (Sec 9.1) and extended (Sec 9.2) sources for all four instruments, including important caveats, staring with the cleaned event files obtained from the archive or the output of reprocessing. For additional details, refer to the previous four chapters.

9.1 Point source

SXS

0. Screen out electrical crosstalk events

```
%> ftselect infile='ah100050020sxs_p0px1010_cl.evt[events]' \
outfile=ah100050020sxs_p0px1010_cl2.evt \
expression="(PI>=400)&&((RISE_TIME>=40 && RISE_TIME<=60 && \
ITYPE<4)||(ITYPE==4))"
```

1. Extract the spectrum

The spectrum is extracted from the entire array (except pixel 12), because the PSF for a point source is larger than the SXS array size. HP and MP grades are used; HP only (`GRADE “0:0”`) and HP+MP+MS (`GRADE “0:2”`) are alternative grade selections. Low cut-off rigidity (COR) time intervals may also be excluded. Spectra may also be extracted using *sxsregext* (Section 5.4.2).

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050020sxs_p0px0000_cl2.evt
    XSELECT> filter GRADE “0:1”       # Extract only Hp and Mp events
    XSELECT> filter COLUMN “PIXEL=0:11 13:35” # Exclude cal pixel
    XSELECT> extract spectrum
    XSELECT> save spectrum sxs_src_HpMp.pi
    XSELECT> exit
```

Caveat: For bright sources (count rate exceeding ~2 cnt/s/pixel), the fraction of Hp/Mp events becomes significantly less than unity. The task *sxsbranch* may be used to estimate the grade branching ratios.

Caveat: If users select certain pixels based on the SKY coordinate, *sxsregext* should be used. This task also creates an exposure map; however, generally this should be regenerated (Step 4).

2. Generate a Non X-ray background spectrum *(not currently relevant)*

```
%>sxsnxbgen infile=ah100050020sxs_p0px1010_cl2.evt \
ehkfile=ah100050020.ehk  regfile=NONE  innxbfile=nxb.fits \
innxbehk=nxb.ehk outpifile=sxs_src_nxb.pi \
sortcol="COR2" sortbin="0,4,5,6,7,8,9,10,11,12,13,99" \
pixels=“0-11,13-35” expr= “(ITYPE==0||ITYPE==1)&&(RISE_TIME>=40&& \
RISETIME<=60)”
```

3. Create an SXS response file

The medium-sized RMF file that includes gaussian core and low-energy exponential tail components is constructed for HP and MP events and the entire array, excluding pixel 12.

```
%> sxsmkrmf infile=ah100050020sxs_p0px1010_cl2.evt \
outfile=sxs_src_HpMp.rmf resolist=0,1 regmode=det \
regionfile=allpixelsNot12_region_SXS_det.reg whichrmf=m
```

4. Create an SXS exposure map

The exposure map generator, *ahexpmap*, should first be run with a fine attitude binning (‘delta=0.25’ and ‘numphi=4’), which is gradually coarsened until every bin has a substantial value in the ‘FRACTION’ column in the ‘OFFAXISHIST’ extension. Here it is assumed that the attitude is sufficiently stable that a single bin is adequate. All files are assumed to be in same working directory.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz \
gtifile=ah100050020sxs_p0px1010_cl2.evt instrume=SXS badimgfile=NONE \
pixgtifile=ah100050020sxs_px1010_exp.gti.gz outfile=sxs_src.expo \
outmaptype=EXPOSURE delta=20 numphi=1 stopsys=SKY instmap=CALDB \
qefile=CALDB contamifile=CALDB vigfile=CALDB obffile=CALDB \
fwfile=CALDB gvfile=CALDB maskcalsrc=yesfwtype=DEFAULT specmode=MONO \
specfile=spec.fits specform=FITS energy=1.5 evperchan=DEFAULT \
abund=1 cols=0 covfac=1
```

5. Create an SXS point source ARF file

The 35-pixel DET coordinate region file (which may be generated using *sxsregext*), and exposure map and RMF files created above, are input here. All files are assumed to be in same working directory. If the number of attitude bins (rows in the exposure map ‘OFFAXISHIST’ extension) is large, the value of ‘numphoton’ may be reduced to speed up the computation.

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxs_ptsrc_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXS dattfile=NONE filtoffsetfile=NONE \
emapfile=sxs_src.expo gatevalvefile=CALDB sampling=1 regmode=DET \
regionfile=allpixelsNot12_region_SXS_det.reg  sourcetype=POINT  \
rmffile=sxs_src_HpMp.rmf  erange="0.5 17.0 0 0" \
outfile=sxs_src_ptsrc.arf numphoton=300000 minphoton=1 \
teldeffile=CALDB qefile=CALDB contamifile=CALDB obffile=CALDB \
fwfile=CALDB gatevalvefile=CALDB onaxisffile=CALDB onaxiscfile=CALDB \
mirrorfile=CALDB obstructfile=CALDB frontreffile=CALDB \
backreffile=CALDB pcolreffile=CALDB scatterfile=CALDB
imgfile=NONE
```

6. Run *grppha* to bin the spectrum using 2 eV bins, and link the spectrum to the RMF, ARF, and NXB files.

```
%> grppha
    grppha> sxs_src_HpMp.pi
    grppha> sxs_src_HPMPgrp.pi
    grppha> group 0 32767 4           # Group into 4 eV channels
    grppha> chkey respfile sxs_src_HpMp.rmf
    grppha> chkey ancrfile sxs_src_ptsrc.arf
    grppha> chkey backfile sxs_src_nxb.pi # If NXB spectrum available
    grppha> exit
```

7. Analyzing the spectrum using XSPEC.

```
    %> xspec
    XSPEC> data 1 sxs_src_HPMPgrp.pi
```

SXI

1. Extract the spectrum

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050020sxi_p0100004b0_cl.evt
    XSELECT> filter STATUS[2]==b0            # Exclude cal sources
    XSELECT> extract image                   # Extract image
    XSELECT> plot image            # display the image on ds9
```

At this point a *ds9* window opens, and a circular extraction region (recommended radius is 2.5 arcmin) sxi_src_ptsrc.reg can be made and saved in SAOImage format. A background region, sxi_bkg_ptsrc.reg, may also be defined, e.g. an annulus centered on the source or off source region.

Now, extract and save spectra using the region files that were just defined.

```
    XSELECT> filter region sxi_src_ptsrc.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum sxi_src.pi
    XSELECT> clear region
    XSELECT> filter region sxi_bkg_ptsrc.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum sxi_bkg.pi
```

Caveat: For bright sources where pile-up may be significant, it is useful to estimate this effect using the *pileest* task. If the pile-up is indeed significant, the affected region should be removed from spectral analysis. The detailed procedure and an example command for *pileest* are given in Section 6.4.1.

2. Create an SXI response file

Run *sxirmf* using the source spectrum file saved in the previous step.

```
%> sxirmf infile=sxi_src.pi outfile=sxi_src.rmf
```

3. Create an SXI exposure map

Here it is assumed that the attitude is sufficiently stable that a single bin is adequate (see SXS section above). All files are assumed to be in same working directory.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz  \
gtifile=ah100050020sxi_p0100004b0_cl.evt instrume=SXI  \
badimgfile=ah100050020sxi_p0100004b0.bimg.gz  \
pixgtifile=ah100050020sxi_a0100004b0.fpix.gz  outfile=sxi_src.expo  \
outmaptype=EXPOSURE delta=20  numphi=1 stopsys=SKY instmap=CALDB \
qefile=CALDB contamifile=CALDB  vigfile=CALDB obffile=CALDB \
fwfile=CALDB gvfile=CALDB maskcalsrc=yes  fwtype=DEFAULT \
specmode=MONO specfile=spec.fits specform=FITS  energy=1.5 \
evperchan=DEFAULT abund=1 cols=0 covfac=1
```

4. Create an SXI ARF file

The SKY coordinate extraction region file, and exposure map and RMF files created above, are input here. All files are assumed to be in same working directory. The value of ‘numphoton’ may need to be reduced to speed up the computation (see SXS section above).

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxi_ptsrc_evt.fits  \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXI dattfile=NONE filtoffsetfile=NONE emapfile=sxi_src.expo \
gatevalvefile=CALDB sampling=1 regmode=SKY \
regionfile=sxi_src_ptsrc.reg   sourcetype=POINT rmffile=sxi_src.rmf \
erange="0.3 17.0 0 0" outfile=sxi_src_ptsrc.arf numphoton=300000  \
minphoton=1 teldeffile=CALDB qefile=CALDB contamifile=CALDB  \
onaxisffile=CALDB \ onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB backreffile=CALDB  \
pcolreffile=CALDB scatterfile=CALDB imgfile=NONE
```

5. Run *grppha* to bin the spectrum, and link the spectrum to the RMF, ARF, and NXB files.

```
 grppha
    grppha> sxi_src.pi
    grppha> sxi_src_grp.pi
    grppha> gr min 50
    grppha> chkey respfile sxi_src.rmf
    grppha> chkey ancrfile sxi_src_ptsrc.arf
    grppha> chkey backfile sxi_bkg.pi
    grppha> exit
```

6. Analyze the spectrum using XSPEC.

```
    %> xspec
    XSPEC> data 1 sxi_src_grp.pi
```

HXI

The steps below are for HXI1, but also may be applied to HXI2 data by replacing "HXI1" and "hx1" by "HXI2" and "hx2", respectively, in what follows.

1. Extract the spectrum

Extraction of spectra for the HXI is similar to that of the SXI if there is only one bright source in the smaller HXI FoV. However if multiple sources are in the field, no background region is available and the extended source procedure described below should be followed for this step. Low cut-off rigidity (COR) time intervals may also be excluded.

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050020hx1_p0camrec_cl.evt.gz
    XSELECT> plot image              # display the image on ds9
```

At this point a *ds9* window opens, and a circular extraction region (recommended radius is 3.0 arcmin) hxi_src_ptsrc.reg can be made and saved in SAOImage format. A background region, hxi_bkg_ptsrc.reg, may also be defined, e.g. an annulus centered on the source or off source region.

Caveat: Even for a point source, any background spectrum from the same field includes some source photons and is an overestimate – an effect that increases with source brightness.

Now, extract and save spectra using the region files that were just defined.

```
    XSELECT> filter region hxi_src_ptsrc.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum hx1_src.pi
    XSELECT> clear region
    XSELECT> filter region hxi_bkg.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum hx1_bkg.pi
```

2. Apply dead time correction to the source and background spectra

```
%> hxisgddtime infile=ah100050020hx1_p0camrecpse_cl.evt.gz \
inlcfile=none inspecfile=hx1_src.pi outlcfile=none \
outfile=hx1_src_dtcor.pi gtifile=ah100050020hx1_p0camrec_cl.evt.gz

%> hxisgddtime infile=ah100050020hx1_p0camrecpse_cl.evt.gz \
inlcfile=none inspecfile=hx1_bkg.pi outlcfile=none \
outfile=hx1_bkg_dtcor.pi gtifile=ah100050020hx1_p0camrec_cl.evt.gz
```

3. Create an HXI1 exposure map

Here it is assumed that the attitude is sufficiently stable that a single bin is adequate (see SXS section above). All files are assumed to be in same working directory.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz \
gtifile=ah100050020hx1_p0camrec_cl.evt.gz instrume=HXI1 \
badimgfile=NONE pixgtifile=NONE outfile=hxi_src.expo \
outmaptype=EXPOSURE delta=20.0 numphi=1 stopsys=SKY instmap=CALDB \
qefile=CALDB contamifile=CALDB vigfile=CALDB obffile=CALDB \
fwfile=CALDB gvfile=CALDB maskcalsrc=yes fwtype=DEFAULT \
specmode=MONO specfile=spec.fits specform=FITS energy=10.0 \
evperchan=DEFAULT abund=1 cols=0 covfac=1
```

4. Create an HXI1 response file

Unlike the SXI and SXS, RMF and ARF files for the HXI are not created separately, but rather combined into a single RSP file. The SKY coordinate extraction region file, and exposure map, created above are input here. The value of 'numphoton' may need to be reduced to speed up the computation (see SXS section above). All files are assumed to be in same working directory.

```
%>aharfgen xrtevtfile=raytrace_ah100050020hx1_ptsrc_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=HXI1 emapfile=hxi_src.expo dattfile=ah100050020hx1.att.gz \
regmode=SKY regionfile=hxi_src_ptsrc.reg  sampling=120 \
sourcetype=point erange="4.0 80.0 0 0" outfile=hxi_src_ptsrc.rsp \
filtoffsetfile=ah100050020hx1_cms.fits.gz numphoton=10000 \
minphoton=1 teldeffile=CALDB qefile=CALDB rmffile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB
```

5. Run *grppha* to bin the spectrum, and link the spectrum to the RSP and NXB files.

```
%> grppha
    grppha> hx1_src_dtcor.pi
    grppha> hx1_src_grp.pi
    grppha> gr min 50
    grppha> chkey respfile hx1_src_ptsrc.rsp
    grppha> chkey backfile hx1_bkg_dtcor.pi
    grppha> exit
```

5. Analyze the spectrum using XSPEC.

```
%> xspec
    XSPEC> data 1 hx1_src_grp.pi
```

HXI – Multiple Sequences

Some extra steps are required when the events for a single observation are split into multiple observation sequences, as in the following HXI1 example where the three sequences are ah100050010, ah100050020, and ah100050030.

1. Extract the spectrum

The same extraction regions used in the previous section are used here.

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050010hx1_p0camrec_cl.evt.gz
    XSELECT> read event ah100050020hx1_p0camrec_cl.evt.gz
    XSELECT> read event ah100050030hx1_p0camrec_cl.evt.gz
    XSELECT> filter region hxi_src_ptsrc.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum hx1_src.pi
    XSELECT> clear region
    XSELECT> filter region hxi_bkg.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum hx1_bkg.pi
```

2. Apply dead time correction to the spectra

a) Merge the pseudo-event files, including their GTI extensions

```
%>ftmerge 'ah100050010hx1_p0camrecpse_cl.evt.gz, \
ah100050020hx1_p0camrecpse_cl.evt.gz, \
ah100050030hx1_p0camrecpse_cl.evt.gz' \
ah1000500ALL0hx1_p0camrecpse_cl.evt
```

```
%>ahgtigen infile=NONE outfile=ah1000500ALL0hx1_p0camrecpse_cl.gti \
gtifile=@ah10005001230hx1_p0camrecpse_cl.gti.lst gtiexpr=NONE \
mergegti=OR
```

where ah10005001230hx1_p0camrecpse_cl.gti.lst is a text file listing all GTI extensions:

```
ah100050010hx1_p0camrecpse_cl.evt.gz+2
ah100050020hx1_p0camrecpse_cl.evt.gz+2
ah100050030hx1_p0camrecpse_cl.evt.gz+2
```

```
%>ftdelhdu 'ah1000500ALL0hx1_p0camrecpse_cl.evt[GTI]' none confirm=YES
```

```
%>ftappend 'ah1000500ALL0hx1_p0camrecpse_cl.gti[GTI]' \
ah1000500ALL0hx1_p0camrecpse_cl.evt
```

b) Merge the event file GTI extensions

```
%>ahgtigen infile=NONE outfile=ah1000500ALL0hx1_p0camrec_cl.gti \
gtifile=@ah10005001230hx1_p0camrec_cl.gti.lst gtiexpr=NONE \
mergegti=OR
```

where ah10005001230hx1_p0camrec_cl.gti.lst is a text file listing all GTI extensions:

ah100050010hx1_p0camrec_cl.evt.gz+2
ah100050020hx1_p0camrec_cl.evt.gz+2
ah100050030hx1_p0camrec_cl.evt.gz+2

```
%>fthedit ah1000500ALL0hx1_p0camrec_cl.gti INSTRUME a HXI1
%>fthedit ah1000500ALL0hx1_p0camrec_cl.gti DETNAM a CAMERA
```

(c) Apply the deadtime correction to the merged source and background spectra, using the merged pseudo-event and event GTI files:

```
%>hxisgddtime infile=ah1000500ALL0hx1_p0camrecpse_cl.evt \
inlcfile=NONE inspecfile=hx1_src.pi outlcfile=NONE
\outfile=hxi_src_dtcor.pigtifile=ah1000500ALL0hx1_p0camrec_cl.gti

%>hxisgddtime infile=ah1000500ALL0hx1_p0camrecpse_cl.evt \
inlcfile=NONE inspecfile=hx1_bkg.pi outlcfile=NONE \
outfile=hx1_dtcor_bkg.pi gtifile=ah1000500ALL0hx1_p0camrec_cl.gti
```

3. Create an HXI1 exposure map

(a) Merge the ehk, delta- attitude, and CAMS offset files from each sequence

```
%>ftmerge'ah100050010.ehk.gz,ah100050020.ehk.gz,ah100050030.ehk.gz' \
ah1000500ALL0.ehk

%>ftmerge \
'ah100050010hx1.att.gz,ah100050020hx1.att.gz,ah100050030hx1.att.gz' \
ah1000500ALL0hx1.att

%>ftmerge 'ah100050010hx1_cms.fits.gz,ah100050020hx1_cms.fits.gz, \
ah100050030hx1_cms.fits.gz' ah1000500ALL0hx1_cms.fits
```

(b) Generate an HXI1 exposure map

```
%>ahexpmap ehkfile=ah1000500ALL0.ehk \
gtifile=ah1000500ALL0hx1_p0camrec_cl.gti instrume=HXI1 \
badimgfile=NONE pixgtifile=NONE outfile=hxi_src.expo \
outmaptype=EXPOSURE delta=20.0 numphi=1 stopsys=SKY instmap=CALDB \
qefile=CALDB contamifile=CALDB vigfile=CALDB obffile=CALDB \
fwfile=CALDB gvfile=CALDB maskcalsrc=yes fwtype=DEFAULT \
specmode=MONO specfile=spec.fits specform=FITS energy=10.0 \
evperchan=DEFAULT abund=1 cols=0 covfac=1
```

4. Create an HXI1 response file

```
%>aharfgen xrtevtfile=raytrace_ah1000500ALL0hx1_p0camrec.fits \
source_ra=278.38824, source_dec=-10.570683 telescop=HITOMI \
instrume=HXI1 emapfile=hxi_src.expo dattfile=ah1000500ALL0hx1.att \
regmode=SKY regionfile=hxi_src_ptsrc.reg  sampling=120 \
sourcetype=point erange="4.0 80.0 0 0" outfile=hxi_src_ptsrc.rsp \
filtoffsetfile=ah1000500ALL0hx1_cms.fits numphoton=10000 minphoton=1 \
teldeffile=CALDB qefile=CALDB rmffile=CALDB onaxisffile=CALDB \
onaxiscfile=CALDB mirrorfile=CALDB obstructfile=CALDB \
frontreffile=CALDB pcolreffile=CALDB scatterfile=CALDB
```

5. Run *grppha* to bin the spectrum, and link the spectrum to the RSP and NXB files.

```
%> grppha
    grppha> hx1_src_dtcor.pi
    grppha> hx1_src_grp.pi
    grppha> gr min 50
    grppha> chkey respfile hx1_src_ptsrc.rsp
    grppha> chkey backfile hx1_bkg_dtcor.pi
    grppha> exit
```

5. Analyze the spectrum using XSPEC.

```
%> xspec
    XSPEC> data 1 hx1_src_grp.pi
```

SGD

The SGD system consists of two detectors, SGD1 and SGD2, each with three Compton cameras, CC1, CC2, and CC3. The steps below are for the combined SGD1 CC1, CC2, CC3 data, but also may be applied to SGD2 data by replacing “SGD1” and “sg1” by “SGD2” and “sg2”, respectively, in what follows. Because the SGD is not an imaging instrument, lightcurves and spectra are always extracted from the entire detector.

1. Extract the spectra from each CC. Low cut-off rigidity (COR) time intervals may also be excluded.

CC1 spectrum:

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event “ah100050020sg1_p0cc1rec_cl.evt”
    XSELECT> extract spectrum
    XSELECT> save spectrum sg1_cc1_src.pi
```

Repeat the above for CC2 and CC3.

Caveat: Currently, there is no standard way to create background spectra.

2. Apply the deadtime correction to the spectrum for each Compton Camera

CC1 spectrum:

```
%> hxisgddtime infile=ah100050020sg1_p0cc1recpse_cl.evt \
inlcfile=none inspecfile=sg1_cc1_src.pi outlcfile=none \
outfile=sg1_cc1_src_dtcor.pi gtifile=ah100050020sg1_p0cc1rec_cl.evt
```

Repeat the above for CC2 and CC3.

3. Sum the deadtime-corrected spectra and add the necessary keywords read from the header of any of the individual spectra (identified using *fkeyprint*), and with the EXPOSURE set to the average of the three individual spectra.

```
%>mathpha
expr=sg1_cc1_src_dtcor.pi+sg1_cc2_src_dtcor.pi+sg1_cc3_src_dtcor.pi
units=C outfil=sg1_ccALL_src_dtcor.pi exposure=2962.33  areascal=%
backscal=% ncomments=0

%>fthedit sg1_ccALL_src_dtcor.pi+1 RA_NOM a 278.384889916054
%>fthedit sg1_ccALL_src_dtcor+1 DEC_NOM a -10.5700436450391
%>fthedit sg1_ccALL_src_dtcor.pi+1 PA_NOM a 10.5700436450391
%>fthedit sg1_ccALL_src_dtcor.pi+1 DATE-OBS a 2016-03-
21T14:42:41.359375
```

3. Generate the SGD1 response files for each CC

As with the HXI, RMF and ARF files for the SGD are not created separately, but rather combined into a single RSP file. The response generator obtains the mean pointing from the input, and any of the spectral files created above may be used. As with the other ARF/RSP generators the source coordinates must also be input.

```
sgdarfgen infile=sg1_cc1_src_dtcor.pi \
rspfile="$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc1_20140101v001.r
sp,$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc2_20140101v001.rsp,$CA
LDB/data/hitomi/sgd/cpf/response/ah_sg1_cc3_20140101v001.rsp" \
outfile=outrsp ra=278.38824  dec=-10.570683 sgdid=1 ccid=0
```

4. Combine the RSP files

```
addrmf outrsp_sgd1_cc1.rsp,outrsp_sgd1_cc2.rsp,outrsp_sgd1_cc3.rsp
1.0,1.0,1.0 ah_sg1_ccALL.rsp
```

5. Run *grppha* to bin spectrum and link the spectrum to rsp.

```
%> grppha
    grppha> sg1_ccall_src_dtcor.pi
    grppha> sg1_ccall_src_grp.pi
    grppha> gr min 50
    grppha> chkey respfile ah_sg1_ccALL.rsp
    grppha> exit
```

6. Analyzing the spectrum using XSPEC.

```
%> xspec
    XSPEC> data 1 sg1_ccall_src_grp.pi
```

SGD – Multiple Sequences

Some extra steps are required when the events for a single observation are split into multiple observation sequences, as in the following SGD1 example where the three sequences are ah100050020, ah100050030, and ah100050040.

1. Extract the combined spectrum for each camera

The same extraction regions used in the previous section are used here.

```
    %> xselect
     XSELECT> xsel
     XSELECT> read events ah100050020sg1_p0cc1rec_cl.evt.gz
     XSELECT> read events ah100050030sg1_p0cc1rec_cl.evt.gz
     XSELECT> read events ah100050040sg1_p0cc1rec_cl.evt.gz
     XSELECT> extract spectrum
     XSELECT> save spectrum sg1_cc1_src.pi
     XSELECT> clear all
     XSELECT> read events ah100050020sg1_p0cc2rec_cl.evt.gz
     XSELECT> read events ah100050030sg1_p0cc2rec_cl.evt.gz
     XSELECT> read events ah100050040sg1_p0cc2rec_cl.evt.gz
     XSELECT> extract spectrum
     XSELECT> save spectrum sg1_cc2_src.pi XSELECT> clear all
     XSELECT> read events ah100050020sg1_p0cc3rec_cl.evt.gz
     XSELECT> read events ah100050030sg1_p0cc3rec_cl.evt.gz
     XSELECT> read events ah100050040sg1_p0cc3rec_cl.evt.gz
     XSELECT> extract spectrum
     XSELECT> save spectrum sg1_cc3_src.pi
```

2. Apply dead time correction to the spectra

a) Merge the pseudo-event files, including their GTI extensions

```
%>ftmerge 'ah100050020sg1_p0cc1recpse_cl.evt.gz,
ah100050030sg1_p0cc1recpse_cl.evt.gz, \
ah100050040sg1_p0cc1recpse_cl.evt.gz' \
ah1000500ALL0sg1_p0cc1recpse_cl.evt

%>ahgtigen infile=NONE outfile=ah1000500ALL0sg1_p0cc1recpse_cl.gti
gtifile=@ah10005002340sg1_p0cc1recpse_cl.gti.lst gtiexpr=NONE
mergegti=OR
```

where `ah10005002340sg1_p0cc1recpse_cl.gti.lst` is a text file listing all GTI extensions:

../100050020/sgd/event_cl/ah100050020sg1_p0cc1recpse_cl.evt.gz+2
../100050030/sgd/event_cl/ah100050030sg1_p0cc1recpse_cl.evt.gz+2
../100050040/sgd/event_cl/ah100050040sg1_p0cc1recpse_cl.evt.gz+2

```
%>ftdelhdu 'ah1000500ALL0sg1_p0cc1recpse_cl.evt[GTI]' none confirm=YES

%>ftappend 'ah1000500ALL0sg1_p0cc1recpse_cl.gti[GTI]' \
ah1000500ALL0sg1_p0cc1recpse_cl.evt

%>fthedit ah1000500ALL0sg1_p0cc1recpse_cl.evt+2 INSTRUME a SGD1
%>fthedit ah1000500ALL0sg1_p0cc1recpse_cl.evt+2 DETNAM a CC1ftdelhdu
'ah1000500ALL0hx1_p0camrecpse_cl.evt[GTI]' none confirm=YES
```

b) Merge the event file GTI extensions

```
%>ahgtigen infile=NONE outfile=ah1000500ALL0sg1_p0cc1rec_cl.gti \
gtifile=@ah10005002340hx1_p0cc1rec_cl.gti.lst gtiexpr=NONE \
mergegti=OR
```

where ah10005002340hx1_p0camrec_cl.gti.lst is a text file listing all GTI extensions:

ah100050020sg1_p0cc1rec_cl.evt.gz+2
ah100050030sg1_p0cc1rec_cl.evt.gz+2
ah100050040sg1_p0cc1rec_cl.evt.gz+2

```
%>fthedit ah1000500ALL0hx1_p0camrec_cl.gti INSTRUME a SGD1
%>fthedit ah1000500ALL0hx1_p0camrec_cl.gti DETNAM a CC!
```

(c) Apply the deadtime correction to the merged source spectrum, using the merged pseudo-event and event GTI files:

```
%>hxisgddtime infile=ah1000500ALL0sg1_p0cc1recpse_cl.evt \
inlcfile=NONE inspecfile=sg1_cc1_src.pi outlcfile=NONE \
outfile=sg1_cc1_src_dtcor.pi gtifile=ah1000500ALL0sg1_p0cc1rec_cl.gti
```

Repeat the above steps (a)-(c) for CC2 and CC3.

(d) Add the individual spectra, setting the EXPOSURE keyword to the average of the three individual spectra.

```
mathpha \
expr=sg1_cc1_src_dtcor.pi+sg1_cc2_src_dtcor.pi+sg1_cc3_src_dtcor.pi \
units=C outfil=sg1_ccALL_src_dtcor.pi \
exposure=42970.67 areascal=% backscal=% ncomments=0
```

3. Generate the SGD1 response files for each CC for each sequence

```
sgdarfgen infile=sg1_p0cc1rec_dtime.pi
rspfile="$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc1_20140101v001.r
sp,$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc2_20140101v001.rsp,$CA
LDB/data/hitomi/sgd/cpf/response/ah_sg1_cc3_20140101v001.rsp"
outfile=outrsp_100050020 ra=278.38824 dec=-10.570683 sgdid=1 ccid=0
clobber=yes

sgdarfgen infile=sg1_p0cc1rec_dtime.pi
rspfile="$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc1_20140101v001.r
sp,$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc2_20140101v001.rsp,$CA
LDB/data/hitomi/sgd/cpf/response/ah_sg1_cc3_20140101v001.rsp"
outfile=outrsp_100050030 ra=278.38824 dec=-10.570683 sgdid=1 ccid=0
clobber=yes

sgdarfgen infile=sg1_p0cc1rec_dtime.pi
rspfile="$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc1_20140101v001.r
sp,$CALDB/data/hitomi/sgd/cpf/response/ah_sg1_cc2_20140101v001.rsp,$CA
LDB/data/hitomi/sgd/cpf/response/ah_sg1_cc3_20140101v001.rsp"
outfile=outrsp_100050040 ra=278.38824 dec=-10.570683 sgdid=1 ccid=0
clobber=yes
```

4. Combine the RSP files

```
addrmf outrsp_100050050_sgd1_cc1.rsp,outrsp_100050050_sgd1_cc2.rsp, \
outrsp_100050050_sgd1_cc3.rsp,outrsp_100050040_sgd1_cc1.rsp, \
outrsp_100050040_sgd1_cc2.rsp,outrsp_100050040_sgd1_cc3.rsp, \
outrsp_100050030_sgd1_cc2.rsp,outrsp_100050030_sgd1_cc3.rsp, \
outrsp_100050030_sgd1_cc1.rsp,outrsp_100050020_sgd1_cc1.rsp, \
outrsp_100050020_sgd1_cc2.rsp,outrsp_100050020_sgd1_cc3.rsp \
0.02016294,0.02016294,0.02016294,0.37231866,0.37231866,0.37231866, \
0.54091513,0.54091513,0.54091513,0.06660327,0.06660327,0.06660327 \
ah_sg1_ccALL.rsp
```

5. Run *grppha* to bin spectrum and link the spectrum to rsp.

```
%> grppha
    grppha> sg1_ccall_src_dtcor.pi
    grppha> sg1_ccall_src_grp.pi
    grppha> gr min 50
    grppha> chkey respfile ah_sg1_ccALL.rsp
    grppha> exit
```

6. Analyzing the spectrum using XSPEC.

```
%> xspec
    XSPEC> data 1 sg1_ccall_src_grp.pi
```

9.2 Extended source

9.2.1 Imaging Analysis

In this section, exposure-corrected images are created for each detector to account for the position-dependent effects of vignetting, contamination, filter transmission, quantum efficiency, etc.

SXS

1. Extract the SXS image

Using XSELECT, one can extract and save a FITS image in the desired energy band, here 0.5-2 keV, using events of all grades (ITYPE).

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050020sxs_p0px0000_cl2.evt
    XSELECT> filter GRADE “0:4”       # Extract Hp, Mp, Ms, Lp, Ls
    XSELECT> filter COLUMN “PIXEL=0:11 13:35” # Exclude cal pixel
    XSELECT> filter pha_cutoff 1000 4000
    XSELECT> extract image
    XSELECT> save image sxs_soft.img
```

2. Create the flat field image

The following command generates a flat-field image for an energy of 1 keV.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz  \
gtifile=ah100050020sxs_p0px1010_cl2.evt instrume=SXS badimgfile=NONE \
pixgtifile=ah100050020sxs_px1010_exp.gti.gz \
outfile=sxs_ffield1keV_soft.img outmaptype=EFFICIENCY delta=0.25 \
numphi=4 stopsys=SKY instmap=CALDB qefile=CALDB contamifile=CALDB \
vigfile=CALDB obffile=CALDB fwfile=CALDB gvfile=CALDB maskcalsrc=yes \
fwtype=DEFAULT specmode=MONO specfile=spec.fits specform=FITS \
energy=1.0 evperchan=DEFAULT abund=1 cols=0 covfac=1
```

The following command generates a flat-field image for the mean energy of the input spectrum created above.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz  \
gtifile=ah100050020sxs_p0px1010_cl2.evt instrume=SXS badimgfile=NONE \
pixgtifile=ah100050020sxs_px1010_exp.gti.gz \
outfile=sxs_ffield1keV_soft.img outmaptype=EFFICIENCY delta=0.25 \
numphi=4 stopsys=SKY instmap=CALDB qefile=CALDB contamifile=CALDB \
vigfile=CALDB obffile=CALDB fwfile=CALDB gvfile=CALDB maskcalsrc=yes \
fwtype=DEFAULT specmode=SPEC specfile=sxs_src_HpMp.pi specform=FITS \
energy=1.0 evperchan=DEFAULT abund=1 cols=0 covfac=1
```

3. Create the exposure-corrected image using the flatfield

The flatfield is applied by dividing the raw count rate image by the efficiency map per pixel created above.

```
%> ftimgcalc outfile=sxs_corrected.img expr=a/b  a=sxs_soft.img \
b=sxs_ffield1keV_soft.img
```

SXI

1. Extract the SXI image

Using XSELECT, one can extract and save a FITS image in the desired energy band, here 0.5-2 keV.

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050020sxi_p0100004b0_cl.evt
    XSELECT> filter STATUS[2]==b0              # Exclude cal sources
    XSELECT> filter pha_cutoff 84 333
    XSELECT> extract image
    XSELECT> save image sxi_soft.img
```

2. Create the flat field image

The following command generates a flat-field image for an energy of 1 keV.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz  \
gtifile=ah100050020sxi_p0100004b0_cl.evt instrume=SXI  \
badimgfile=ah100050020sxi_p0100004b0.bimg.gz  \
pixgtifile=ah100050020sxi_a0100004b0.fpix.gz  \
outfile=sxs_ffield1keV_soft.img  outmaptype=EFFICIENCY  delta=0.25  \
numphi=4 stopsys=SKY instmap=CALDB qefile=CALDB contamifile=CALDB  \
vigfile=CALDB obffile=CALDB fwfile=CALDB gvfile=CALDB maskcalsrc=yes \
fwtype=DEFAULT specmode=MONO specfile=spec.fits specform=FITS \
energy=1.0 evperchan=DEFAULT abund=1 cols=0 covfac=1
```

The following command generates a flat-field image for the mean energy of the input spectrum created above.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz  \
gtifile=ah100050020sxi_p0100004b0_cl.evt instrume=SXI  \
badimgfile=ah100050020sxi_p0100004b0.bimg.gz  \
pixgtifile=ah100050020sxi_a0100004b0.fpix.gz  \
outfile=sxs_ffield1keV_soft.img  outmaptype=EFFICIENCY  delta=0.25  \
numphi=4 stopsys=SKY instmap=CALDB qefile=CALDB contamifile=CALDB  \
vigfile=CALDB obffile=CALDB fwfile=CALDB gvfile=CALDB maskcalsrc=yes \
fwtype=DEFAULT specmode=SPEC specfile=sxi_src.pi  specform=FITS \
energy=1.0 evperchan=DEFAULT abund=1 cols=0 covfac=1
```

3. Create the exposure-corrected image using the flatfield

```
%> ftimgcalc outfile=sxi_corrected.img expr=a/b  a=sxi_soft.img \
b=sxi_ffield1keV_soft.img
```

HXI

1. Extract the HXI image

Using XSELECT, one can extract and save a FITS image in the desired energy band, here 5.0 - 20.0 keV.

```
     %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050020hx1_p0camrec_cl.evt
    XSELECT> filter pha_cutoff 50 200
    XSELECT> extract image
    XSELECT> save image hx1_5to20.img
```

2. Create the flat field image

Because the motion of the EOB must be accounted for, the HXI flat field calculation is a two-step process. First the exposure map is created.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz \
gtifile=ah100050020hx1_p0camrec_cl.evt.gz instrume=HXI1 \
badimgfile=NONE pixgtifile=NONE outfile=hxi_src.expo1  \
outmaptype=EXPOSURE delta=0.5 numphi=4 stopsys=SKY instmap=CALDB \
qefile=CALDB contamifile=CALDB vigfile=CALDB obffile=CALDB \
fwfile=CALDB gvfile=CALDB maskcalsrc=yes fwtype=DEFAULT \
specmode=MONO specfile=spec.fits specform=FITS energy=10.0 \
evperchan=DEFAULT abund=1 cols=0 covfac=1
```

Next, the flat-field image is created using the exposure map.

```
%> hxirspeffimg telescop=HITOMI instrume=HXI1 emapfile=hxi_src.expo1 \
xrtevtfile=raytrace_ah100050020hx1_p0camrec.fits onaxisffile=CALDB  \
onaxiscfile=CALDB regionfile=NONE dattfile=ah100050020hx1.att.gz \
stopsys=SKY sampling=40 erange="5.0 20.0" \
filtoffsetfile=ah100050020hx1_cms.fits.gz  \
outflatfile=hx1_flat_5to20.img vigfile=CALDB outmaptype=EFFICIENCY \
qefile=CALDB rmffile=CALDB chatter=2 mode=h outfile=NONE sampling=1
```

3. Create the exposure-corrected image using the flatfield

```
%> ftimgcalc outfile=hx1_corrected.img expr=a/b a=hx1_5to20.img \
b=hx1_flat_5to20.img
```

9.2.2 Spectral Analysis

This section describes how to analyze spectral data of extended sources. Most of the procedures are the same as those for point sources. Note that for a source spectrum extracted from a region with point and extended source components, both point-source (as illustrated above) and extended source (as illustrated below) ARF files can be constructed and separately applied to spectral components associated with the two spatial components.

SXS

For the SXS, the extraction of spectra and generation of rmf files is the same for extended and point sources (Section 5.6.1). Extended-source ARF files may be made for flat or beta-model, distributions, or for an input image (Section 5.6.2). The following creates an ARF for the 2-8 keV input image `image.fits.`

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxs_image_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=SXS dattfile=NONE filtoffsetfile=NONE \
emapfile=sxs_src.expo regmode=DET gatevalvefile=CALDB sampling=1 \
regionfile=allpixelsNot12_region_SXS_det.reg  sourcetype=image \
imgfile=image.fits rmffile=sxs_src_HpMp.rmf \
erange="0.5 17.0 2.0 8.0" outfile=sxs_src_image.arf numphoton=300000 \
minphoton=1 teldeffile=CALDB qefile=CALDB contamifile=CALDB \
obffile=CALDB fwfile=CALDB gatevalvefile=CALDB onaxisffile=CALDB \
onaxiscfile=CALDB mirrorfile=CALDB obstructfile=CALDB \
frontreffile=CALDB backreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB
```

SXI

In the following example, the background spectrum is extracted from a region far from the source region (or even from a separate observation) so that the vignetting is very different, and simultaneous fitting of source and background spectra is recommended. Otherwise (except for the ARF generation) the steps are the same as for a point source.

1. Extract the spectrum

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050020sxi_p0100004b0_cl.evt
    XSELECT> filter STATUS[2]==b0            # Exclude cal sources
    XSELECT> extract image                   # Extract image
    XSELECT> plot image              # display the image on ds9
```

At this point a ds9 window opens, and an extraction region sxi_src_extsrc.reg can be made and saved in SAOImage format. A background region, sxi_bkg_extsrc.reg, may also be defined, e.g. an annulus centered on the source or off source region.

Now, extract and save spectra using the region files that were just defined.

```
    XSELECT> filter region sxi_src_extsrc.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum sxi_src.pi
    XSELECT> clear region
    XSELECT> filter region sxi_bkg_extsrc.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum sxi_bkg.pi
```

2. Generate Non X-ray background (NXB) spectra *(not currently relevant)*

Run the '*sxinxbgen'* task to generate NXB spectra for each region. The region files used for the source spectrum are required in this step.

```
 %> sxinxbgen infile=sxi_src.pi \
 ehkfile=ah100050020.ehk regfile=sxi_src_extsrc.reg  \
 innxbfile=nxb.fits innxbehk=nxb.ehk innxbhk=nxb.hk \
 outpifile=sxi_src_nxb.pi

 %> sxinxbgen infile=sxi_bkg.pi\
 ehkfile=ah100050020.ehk regfile=sxi_bkg_extsrc.reg \
 innxbfile=nxb.fits innxbehk=nxb.ehk innxbhk=nxb.hk \
 outpifile=sxi_bkg_nxb.pi
```

3. Creating response files

```
  %> sxirmf infile=sxi_src.pi outfile=sxi_src.rmf

  %> sxirmf infile=sxi_bkg.pi outfile=sxi_bkg.rmf
```

4. Create the SXI exposure map

This step is identical to that for point sources shown above.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz  \
gtifile=ah100050020sxi_p0100004b0_cl.evt instrume=SXI  \
badimgfile=ah100050020sxi_p0100004b0.bimg.gz  \
pixgtifile=ah100050020sxi_a0100004b0.fpix.gz  outfile=sxi_src.expo  \
outmaptype=EXPOSURE delta=20  numphi=1 stopsys=SKY instmap=CALDB \
qefile=CALDB contamifile=CALDB  vigfile=CALDB obffile=CALDB \
fwfile=CALDB gvfile=CALDB maskcalsrc=yes  fwtype=DEFAULT \
specmode=MONO specfile=spec.fits specform=FITS  energy=1.5 \
evperchan=DEFAULT abund=1 cols=0 covfac=1
```

5. Create an SXI extended source ARF file

This step is identical to that for point sources shown above, except we assume a flat spatial distribution of radius 5 arcminutes for the source ARF, and additionally construct a background ARF for a flat spatial distribution of radius 50 arcminutes.

```
%>aharfgen xrtevtfile=raytrace_ah100050020sxi_extsrc_evt.fits  \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
dattfile=NONE filtoffsetfile=NONE gatevalvefile=CALDB sampling=1 \
instrume=SXI emapfile=sxi_src.expo  regmode=SKY \
regionfile=sxi_src_extsrc.reg   sourcetype=FLATCIRCLE flatradius=5.0 \
rmffile=sxi_src.rmf erange="0.3 17.0 0 0"  outfile=sxi_src_extsrc.arf \
numphoton=300000  minphoton=1 teldeffile=CALDB qefile=CALDB \
contamifile=CALDB  onaxisffile=CALDB \ onaxiscfile=CALDB \
mirrorfile=CALDB obstructfile=CALDB frontreffile=CALDB \
backreffile=CALDB  pcolreffile=CALDB scatterfile=CALDB imgfile=NONE

%>aharfgen xrtevtfile=raytrace_ah100050020sxi_extbkg_evt.fits  \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
dattfile=NONE filtoffsetfile=NONE gatevalvefile=CALDB sampling=1 \
instrume=SXI emapfile=sxi_src.expo  regmode=SKY \
regionfile=sxi_bkg_extsrc.reg   sourcetype=FLATCIRCLE flatradius=50.0 \
rmffile=sxi_bkg.rmf erange="0.3 17.0 0 0"  outfile=sxi_bkg_extsrc.arf \
numphoton=300000  minphoton=1 teldeffile=CALDB qefile=CALDB \
contamifile=CALDB  onaxisffile=CALDB onaxiscfile=CALDB \
mirrorfile=CALDB obstructfile=CALDB frontreffile=CALDB \
backreffile=CALDB  pcolreffile=CALDB scatterfile=CALDB imgfile=NONE
```

6. Run *grppha* to bin the spectrum, and link the spectrum to the RMF, ARF, and NXB files

```
    %> grppha
    grppha> sxi_src.pi
    grppha> sxi_src_grp.pi
    grppha> gr min 50
    grppha> chkey respfile sxi_src.rmf
    grppha> chkey ancrfile sxi_src_extsrc.arf
    grppha> chkey backfile sxi_src_nxb.pi # if available
    grppha> exit

    %> grppha
    grppha> sxi_bkg.pi
    grppha> sxi_bkg_grp.pi
    grppha> gr min 50
    grppha> chkey respfile sxi_bkg.rmf
    grppha> chkey ancrfile sxi_bkg_extsrc.arf
    grppha> chkey backfile sxi_bkg_nxb.pi # if available
    grppha> exit
```

7. Analyze the spectrum using XSPEC.

```
    %> xspec
    XSPEC> data 1:1 sxi_src_grp.pi
    XSPEC> data 2:2 sxi_bkg_grp.pi
```

Note that the NXB has already been subtracted from both source (src) and background (bkg) spectra. The latter should therefore only contain X-ray background components (e.g., local Galactic background, CXB, etc.). The source and background spectra should be simultaneously fitted using appropriate models for the background components (e.g., power law for the CXB).

HXI

HXI analysis of extended sources can be performed similarly to the SXI, i.e, simultaneous fitting of source and background spectra after NXB subtraction.

1. Extract source and background spectra

```
    %> xselect
    XSELECT> xsel
    XSELECT> read event ah100050020hx1_p0camrec_cl.evt
    XSELECT> filter region src.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum hx1_src.pi
    XSELECT> clea region
    XSELECT> filter region bkg.reg
    XSELECT> extract spectrum
    XSELECT> save spectrum hx1_bkg.pi
```

2. Apply dead time correction to the spectra

```
hxisgddtime infile=ah100050020hx1_p0camrecpse_cl.evt.gz \
inlcfile=none inspecfile=hx1_src.pi outlcfile=none \
outfile=hx1_src_dtcor.pi \
gtifile=ah100050020hx1_p0camrec_cl.evt.gz

hxisgddtime infile=ah100050020hx1_p0camrecpse_cl.evt.gz\
inlcfile=none inspecfile=hx1_bkg.pi outlcfile=none \
outfile=hx1_bkg_dtcor.pi \
gtifile=ah100050020hx1_p0camrec_cl.evt.gz
```

3. Generating Non X-ray background (NXB) spectra *(not currently relevant)*

```
 %> hxinxbgen infile=hx1_src_dtcor.pi \
 ehkfile=ah100050020.ehk.gz regfile=src.reg \
 innxbfile=nxb.fits inpsefile=nxb.pse innxbehk=nxb.ehk \
 outpifile=hx1_src_nxb.pi

 %> hxinxbgen infile=hx1_bkg_dtcor.pi \
 ehkfile=ah100050020.ehk regfile=bkg.reg \
 innxbfile=nxb.fits inpsefile=nxb.pse innxbehk=nxb.ehk  \
 outpifile=hx1_bkg_nxb.pi
```

4. Create response files

This step is identical to that for point sources shown above.

```
%>ahexpmap ehkfile=ah100050020.ehk.gz \
gtifile=ah100050020hx1_p0camrec_cl.evt.gz instrume=HXI1 \
badimgfile=NONE pixgtifile=NONE outfile=hxi_src.expo  \
outmaptype=EXPOSURE delta=20.0 numphi=1 stopsys=SKY instmap=CALDB \
qefile=CALDB contamifile=CALDB vigfile=CALDB obffile=CALDB \
fwfile=CALDB gvfile=CALDB maskcalsrc=yes fwtype=DEFAULT \
specmode=MONO specfile=spec.fits specform=FITS energy=10.0 \
evperchan=DEFAULT abund=1 cols=0 covfac=1
```

This step is identical to that for point sources shown above, except we assume a flat spatial distribution of radius 3 arcminutes for the source ARF, and additionally construct a background ARF for a flat spatial distribution of radius 10 arcminutes.

```
%>aharfgen xrtevtfile=raytrace_ah100050020hx1_extsrc_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=HXI1 emapfile=hxi_src.expo dattfile=ah100050020hx1.att.gz \
regmode=SKY regionfile=src.reg  sampling=120 sourcetype=FLATCIRCLE  \
flatradius=3.0 erange="4.0 80.0 0 0" outfile=hxi_src_extsrc.rsp \
filtoffsetfile=ah100050020hx1_cms.fits.gz numphoton=10000 \
minphoton=1 teldeffile=CALDB qefile=CALDB rmffile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB

%>aharfgen xrtevtfile=raytrace_ah100050020hx1_extbkg_evt.fits \
source_ra=278.38824  source_dec=-10.570683 telescop=HITOMI \
instrume=HXI1 emapfile=hxi_src.expo dattfile=ah100050020hx1.att.gz \
regmode=SKY regionfile=bkg.reg  sampling=120 sourcetype=FLATCIRCLE  \
flatradius=10.0 erange="4.0 80.0 0 0" outfile=hxi_bkg_extsrc.rsp \
filtoffsetfile=ah100050020hx1_cms.fits.gz numphoton=10000 \
minphoton=1 teldeffile=CALDB qefile=CALDB rmffile=CALDB \
onaxisffile=CALDB onaxiscfile=CALDB mirrorfile=CALDB \
obstructfile=CALDB frontreffile=CALDB pcolreffile=CALDB \
scatterfile=CALDB
```

5. Run *grppha* to bin the spectrum, and link the spectrum to the RSP and NXB files.

```
    %> grppha
    grppha> hx1_src.pi
```

```
grppha> hx1_src_grp.pi
grppha> gr min 50
grppha> chkey respfile hxi_src_extsrc.rsp
grppha> chkey backfile hx1_src_nxb.pi # if available
grppha> exit

%> grppha
grppha> hx1_bkg.pi
grppha> hx1_bkg_grp.pi
grppha> gr min 50
grppha> chkey respfile hxi_bkg_extsrc.rsp
grppha> chkey backfile hx1_bkg_nxb.pi
grppha> exit
```

7. Analyze the spectrum using XSPEC.

```
%> xspec
XSPEC> data 1:1 hx1_src_grp.pi
XSPEC> data 2:2 hx1_bkg_grp.pi
```

Note that the NXB has already been subtracted from both source (src) and background (bkg) spectra. The latter should therefore only contain X-ray background components (e.g., local Galactic background, CXB, etc.). The source and background spectra should be simultaneously fitted using appropriate models for the background components (e.g., power law for the CXB). Users should refer to the XSPEC manual for details of the simultaneous fitting.

10 APPENDICES

APPENDIX A: FILTERING

Cleaned event files are generated within the pipeline by screening for either GTI or events. Screening for GTI includes several type of GTIs either generated from an expression apply to special files (mkf and ehk files) or calculated by software (mxsgti) or derived directly from telemetry. Screening for any event characteristic is done using an expression applied to specific columns in the FITS event data. Two scripts, *ahgtigen* and *ahscreen* are used in the pipeline and in the ahpipeline for the cleaning of the event file. GTI are either created in the pre-pipeline and send with the FFF data or calculated within the pipeline. The pre-pipeline GTIs cannot be recreated. The pipeline GTIs may be recreated using software & screening expression. The expressions to create GTI or select events are stored in a CALDB file called: ah_gen_select_YYYYMMDDvXXX. Each expression is identified with a label, the instrument and the file to which that expression is applicable. The expressions are applied to the mkf and ehk files to create GTI or directly to the event files (see screening criteria given in each chapter). The GTI type applied to the data are:
a) GTI with general pointing information generated with the FFF. These GTI define the times on source or slew and are created in the pre-pipeline
b) GTI with the telemetry saturation generated with the FFF. These GTI are instrument dependent and are also created in the pre-pipeline
c) GTI derived from instrument-related HK stored in the mkf file. These GTI are created with *ahgtigen* in the pipeline using an expression applicable to the mkf file. The mkf file is created in the pipeline with ahfilter.
d) GTI derived from orbital information stored in the ehk file. These GTI are created with *ahgtigen* in the pipeline using an

expression applicable to the ehk file. The ehk file is created in the pipeline with ahmkehk.
e) GTI from instrument subsystems (applicable to the SXS, SXI). For example, the SXS special GTI exclude data from the MXS source & interval of "lost event". MXS GTIs are created by software (msxtime). Other examples include the lost GTI derived from telemetry and for the SXI, special GTI derived for each of the data mode (sximodegti).

The task *ahgtigen* is script that has two primary functions. It can either create a GTI file using an expression operating on columns stored in a FITS file, or it can merge two or more GTI files. To create a GTI file, the user must specify a time ordered input file to which the expression based on columns present in the file are to be applied. The expression is either specified by the user or is one of the already made expressions present in the CALDB select file. To merge two or more GTI files, the user must input the list of the GTI to be merged using an AND or OR logic. The main input parameter of the *ahgtigen* task are:

- File to apply the expression (mkf or ehk)
- GTI file – (single or multiple)
- Select CALDB file from where the event expression is read and/or user expression
- (labels to read from the CALDB selectfile if expression is from CALDB)
- Merge GTI with and/or (parameter mergegti)

The output is a GTI file derived from either the "create" or the "merge" operation.

The task *ahscreen* is a script that screens a science event file by applying an expression to the event and/or screening event for GTI. *ahscreen* merges GTI with any other that have been input with either a 'OR' or 'AND' logic. The event screening uses the expression either input by the user or found in the CALDB select file. The main input parameter of the *ahscreen* task are:

- Event file to screen
- GTI file – (single or multiple)
- Select CALDB from where the event expression is read or user expression
- (labels to read from the CALDB selectfile if expression is from CALDB)
- Merge GTI with and/or (parameter mergegti),

The output is an event file screened with the final GTI kept in the second extension

Below are three examples on how to use *ahgtigen* and *ahscreen.*

1) **Example 1:** Creating a GTI file from the general mkf file using expression in file "ah_gen_select_20140101v001.fits" labeled SGDSFFA3CC1READ.

```
ahgtigen infile=ah000506304.mkf outfile=ah000506304mkf-SGD1 gtifile=NONE \
gtiexpr=NONE mergegti=AND instrume=SGD1 selectfile=ah_gen_select_20140101v001.fits\
label=SGDSFFA3CC1READ clobber=yes
```

In the file ah_gen_select_20140101v001.fits, the label SGDSFFA3CC1READ is associated to the selection criteria: HKSG1AB001==b1&&HKSG1AB002==b1 (see Chapter 8, Table 8.3)

2) **Example 2:** Create a GTI from the mkf using the CALDB file to read the expression . GTI is created for the SXS with the label PIXELALL.

```
ahgtigen infile=ah000506304.mkf outfile=ah000506304mkf-SXSCALDB \
gtifile=NONE gtiexpr=NONE mergegti=AND instrume=SXS selectfile=CALDB \
label=PIXELALL clobber=yes
```

3) **Example 3:** Running ahscreen using the GTI file created from Example 2 and selecting only **e**vents with High grade (ITYPE=0.)

```
ahscreen infile=sxs.evt outfile=sxs_cl.evt \
gtifile=ah000506304mkf-SXS-CALDB expr='ITYPE=0' mergegti=AND
```

APPENDIX B: IMPORTANT TASKS

This is a summary of all the tasks used in the Hitomi analysis and referenced in this document.

Task Name	Function
General Tasks	
ahgtigen	Create and/or merge GTI files
ahmodhkext	Merge extensions of FITS file into single extension
ahscreen	Screen a science event file
Mission Tasks	
aharfgen	Make an ancillary response function (ARF) file for the SXS or the SXI and a response matrix (RSP) file for the HXI
ahcalctime	Rerun the timing tools on an output directory from the pipeline
ahexpmap	Generate an exposure map for HXI, SXI, and SXS, or a flat field image for SXS and SXI
ahfilter	Generate an EHK file and a MKF filter file
ahgainfit	Calculate the time-dependent energy gain corrections using known calibration lines
ahmkehk	Extract and calculate parameters from orbit and attitude (EHK)
ahmkregion	Create regions files showing the field of view of all Hitomi instruments
ahmktim	Calculate the relation of Time Invariant vs Time and the GTI for when the GPS is on
ahpipeline	HXI/SGD/SXS/SXI reprocessing tool
ahsxtarfgen	Create an ancillary response function (ARF) file for the systems SXS+SXT-S or the SXI +SXT-I accounting for the telescope effective area and detector efficiencies.
ahtime	Calculate times and populate the TIME column for all science and housekeeping data
ahtimeconv	Convert mission time in different formats
ahtrendtemp	Calculate the quartz clock frequency vs temperature relation for the on-board clock
HXI/SGD Tasks	
cams2att	Computes a time-dependent delta-attitude file
cams2det	Calculate offsets and rotations in HXI coordinates using CAMS telemetry input
camssim	(1) Simulates motion of the Extensible Optical Bench (EOB) for testing CAMS-correction. (2) Generates RAW coordinates associated with given HXI ACT coordinates and the simulated motion
hxievtid	Reconstructs HXI events
hxigainfit	Calculate the HXI time-dependent energy gain corrections for events from comparison with known calibration lines
hxinxbgen	Create a Non-X-ray Background (NXB) spectrum for HXI
hxipipeline	HXI reprocessing tool
hxirspeffimg	Creates a combined effective area and spectral response matrix ("RSP" file) and/or a flat-field correction image, for the HXI1 + HXT1 or HXI2 + HXT2 combinations, accounting for the telescope effective area, detector efficiencies, and CAMS motion.
hxisgddtime	Calculate and correct for deadtime HXI and SGD spectra and light-curves
hxisgdexpand	Expand HXI and SGD SFF occurrences to have one signal in each row
hxisgdmerge	Merge slew and pointing raw (unprocessed) event files for the HXI or SGD
hxisgdpha	Calibrates the HXI or SGD PHA for each signal in the SFF event file
hxisgdsff	Converts an HXI or SGD First FITS File (FFF) into the Second FITS File (SFF)
hxisgdshield	Extract light-curves or spectra from shield data of HXI or SGD
sgdarfgen	Calculate SGD transmission ratio for a source on the sky and create ARF file
sgdevtid	Reconstruct SGD events
sgdgainfit	Calculate the SGD time-dependent energy gain corrections for events from comparison with known calibration lines
sgdpipeline	SGD reprocessing tool
SXI Tasks	
sxiflagpix	Flag pixel STATUS for SXI event data
sxigainfit	Calculate the SXI time-dependent energy gain corrections for events from comparison with known calibration lines
sximodegti	Create GTI excluding dead time for each SXI observing
sxinxbgen	Create a Non-X-ray Background (NXB) spectrum for SXI
sxiphas	Merge inner 3x3 and outer 5x5 pulse height columns in SXI event list
sxipi	Calculate pulse invariant (PI) values and assign grades for SXI events

sxipipeline	Calculate pulse invariant (PI) values and assign grades for SXI events
sxirmf	Create an SXI energy Redistribution Matrix File (RMF)
SXS Tasks	
mxsgti	Create MXS GTI files
mxstime	Calculate the times of the MXS start and stop and generates coarse and fine GTI files
sxsanticolc	Extract SXS antico light curve(s) and spectra using optional screening criteria
sxsanticopi	Assign PHA and PI columns to the SXS anticoincidence events
sxsbranch	Compute rates, branching ratios, and effective exposure times for each SXS event grade for an input real or simulated file, or based on a count rate
sxsflagpix	Flag SXS events for antico and MXS event coincidence, temporal proximity, and crosstalk
sxsgain	Calculate the time-dependent energy correction for SXS events from comparison with known calibration lines
sxsmkrmf	Create an SXS RMF file and/or an RSP file for selected SXS pixels and grades
sxsnxbgen	Create a Non-X-ray Background (NXB) spectrum for SXS
sxsperseus	Remove differential gain error and fiducial gain error
sxspha2pi	Calculate the PI for the SXS event files
sxspipeline	SXS reprocessing tool
sxspixgti	Create SXS GTI files
sxsregext	Extract SXS data products from an event file using a region and selection of grades
sxsrmf	Create an SXS RMF file for selected SXS pixels and grades with weighting factors
sxssamcnt	Calculate the local time in the SXS files necessary to assign time
sxsseccor	Correct PHA for secondary events
sxssecid	Associate SXS secondary events to the primary and allow to recalculate event grades
Non-Hitomi Tasks	
Attitude Tasks	
aberattitude	Correct an attitude file for aberration effects
aberposition	Correct coordinates for aberration effects
attconvert	Converts attitude format
coordevt	Convert events from one coordinate system to another
coordpnt	Convert a single point or region file from one coordinate system to another
det2att2	Converts detector coordinate offsets and rotations to attitude quaternions
FTOOLS TIME Tasks	
gtiinvert	Create a GTI file by inverting the time intervals of an input GTI file
gticolconv	Merge or split GTI based on specified 'column' and 'direction'
HEASIM Tasks	
heasim	Multi-mission high-energy astrophysics simulation tool
skyback	Simulates the total, broadband, discrete and diffuse high-energy astrophysical background
HEAGEN Tasks	
arftable	Create an ancillary response function (ARF), or effective area file from the output events from the ray-tracing code *xrtraytrace*, not including any detector efficiencies
eeftable	Create an encircled energy function (EEF) file based on the output history file from the ray-tracing tool *xrtraytrace*
xraytrace	Perform ray-tracing simulations of X-ray telescopes, calculating photon paths, PSF, EEF, and effective area
xrtreftable	Calculate the probability of reflection and transmission of a ray upon a surface, and mass absorption coefficients for mirror surface
barycen	Apply barycenter corrections to X-ray timing data
HEASARC Tasks	
searchflickpix	Search for anomalous 'flickering' pixels in event files from CCD-type detectors

APPENDICE C: List of CALDB files

Filename	Function	Tasks
GEN		
bcf/		
atmsca	Atomic data to calculate the mirror	xraytrace
coldef	Time assignment	ahtime
delay	Time delay for all subsystems	ahtime, msxtime
linefit	Line energies component for the gain fitting (all instruments)	sxsgain, ahgainfit
mkfconf	Configuration file	ahfilter
qclocka/qclockb	Time assignment clock stability	ahmktim
saa	Hold the SAA vertices	ahmkehk
select	Select/screening criteria for event and GTI	ahscreen, ahgtigen
vigncoef	Coefficient or the vignetting function fitting	
Not Hitomi specific		
atomic	Atomic data to calculate the mirror reflectivity	
leapsec	Leap seconds	
rigidity	Rigidity (Suzaku and 2016 version)	
HXI & CAMS		
bcf/		
cams/tempxy	CAMS coefficient to calculate X Y	cams2att
cams/offset	CAMS time offset file	
gain/enecut	Contain the energy cut for top and bottom HXI layers	hxievtid
gain/gain	Gain parameters	hxisgdpha
gain/line	Fluorescence line data for the HXI	hxievtid
instmap/badpix	Bad pixel and threshold of the HXI (2 files)	hxievtid, hxisgdpha
instmap/instmap	HXI FOC/RAW Instrument map for simulation and exposure	
instmap/remap	HXI detector naming component (1 file)	hxievtid, hxisgdpha
mirror/mirror	Description of the HXT mirror components	xraytrace
mirror/reftrans	Description of the HXT mirror reflectivity	xraytrace
mirror/scatter	Description of the HXT mirror scattering	xraytrace
mirror/telarea	Coarse and fine telescope effective area files & baffle description	aharfgen hxirspeffimg
quanteff/lsf	Line spread function to use with QE for the arf generator	hxirrsoeffimg
quanteff/quanteff	Library of QE for the HXI (5 for each HXI: one/layer and one total)	hxirspeffimg
teldef/teldef	Telescope definition file for HXIs & CAMS	coordevt, coordpnt, cams2att, cams2det
cpf/		
response/rmf x layer & total	HXI Response Function (5 for each HXI)	
background/hxi_nxbext	Non X-ray background spectrum for extended source	Simulations
background/hxi_nxbpntl	Non X-ray background spectrum for point source	Simulations
background/hxi_nxbpnts	Non X-ray background spectrum for point source	Simulations
SGD		
bcf/		
gain/gain	Gain parameter for the SGD and CC components	hxisgdpha
gain/line	Fluorescence data in the SGD (one file)	sgdevtid
instmap/badpix	Bad pixel and threshold of the SGD and CC component	
instmap/instmap	Instrument map for simulation	
instmap/remap	Description of the SGD detector component	hxisgdsffa, hxisgdpha
prob/probfov	Probability of FOV for each of the SGD	sgdevtid
prob/probseq	SGD Probability associated to hit sequences	sgdevtid
response/transrat	SGD Instrument transmission ratio (2 files, 3 extensions)	sgdevtid
teldef/teldef	Telescope definition file for SGD1 and SGD2	
cpf/		
response/sg1/2_cc[1-3].rsp	Responses files – 6 RSP (each for SGDn-CCm)	
background/nxb	Non X-Ray background	
SXI		

bcf/		
gain/chtrail	Charge trail SXI	sxipi
gain/cti	Charge transfer inefficiency SXI	sxipi
gain/gain	Gain coefficients SXI	sxipi
gain/pattern	Patters of the 3x3 SXI	sxipi
gain/spth	Split Threshold SXI	sxipi
gain/vtevnodd	Video temperature even and odd SXI pixels	sxipi
instmap/badpix	Bad pixel in the SXI	sxiflagpix
instmap/config	SXI Data Class Configuration	
instmap/instmap	FOC/DET Instrument map for simulation and exposure	
instmap/mask	Mask table for the SXI in ACT & DET coordinates	
mirror/mirror	Description of the SXT-I mirror components	xraytrace
mirror/reftrans	Description of the SXT-I mirror reflectivity	xraytrace
mirror/scatter	Description of the SXT-I mirror scattering	xraytrace
mirror/telarea	Effective area library	aharfgen, ahsxtarfgen
quanteff/quanteff	Quantum efficiency	ahsxtarfgen
quanteff/contami	Contamination for the SXI	
response/rmfparam	Instrumental parameters	sxirmf
teldef/teldef	Telescope definition file SXI (primary +extension)	coordevt, coordpnt
cpf/		
background/nxbpnt	Spectrum of Non X-ray background	
SXS		
bcf/		
gain/gainant	Gain for the antico	Sxsanticopi
gain/gainpix	Gain for the pixel temp dependent	sxsgain, sxspha2pi
gain/scale	Scale of the pixel gain calculated using Pixel 12	
gain/secpulse	SXS Secondary Pulse	
instmap/badpix	Bad pixel	
instmap/confthre	SXS Threshold definitions	
instmap/instmap	FOC/DET Instrument map for simulation and exposure	
instmap/pixmap	SXS pixel position	sxsflagpix
mirror/mirror	Description of the SXT-S mirror components	xraytrace
mirror/reftrans	Description of the SXT-S mirror reflectivity	xraytrace
mirror/scatter	Description of the SXT-S mirror scattering	xraytrace
mirror/telarea	Effective area library	aharfgen, ahsxtarfgen
quanteff/blckfilt	SXS blocking filter transmission	ahsxtarfgen
quanteff/contami	Column density and energy-dependent transmittance of contamination materials	aharfgen
quanteff/fwbe	SXS Be Filter file	ahsxtarfgen
quanteff/fwfe55	SXS Fe 55 Filter file	ahsxtarfgen
quanteff/fwnd	SXS Neutral Density ND25 Filter File	ahsxtarfgen
quanteff/fwpoly	SXS POLYIMIDE Filter file	ahsxtarfgen
quanteff/gatevalv	SXS Gate Valve Calibration File	ahsxtarfgen
quanteff/quanteff	SXS Quantum Efficiency	ahsxtarfgen
response/rmfparam	SXS rmf parameters	sxsrmf, sxsmkrmf
time/coeftime	SXS timing coefficients	sxssamcnt
teldef/teldef	Telescope definition file SXS (primary +extension)	coordevt, coordpnt

APPENDICE D: List of FITS files and important columns in these files

For all observations, the data directory contains the following directories: auxil, hxi, sgd, sxi and sxs. Below are the details of the content of all the instrument directories and their subdirectories (see also Appendix G). In all what follows, OBS_ID is 8-digit observation ID (same as OBS_ID keyword), 1 or 2 for HXI1, SGD1 or HXI2, SGD2 (matches INSTRUME keyword), p or s for pointing or slew (matches OBS_MODE keyword), and 0-9 is file number when data is split into multiple files

1) GENERAL/AUXIL

ah[OBS_ID].tim.gz

Extension	Description
TIME_PACKETS	Cleaned events
TIM_LOOKUP	Good Time Interval for cleaned events

TIME_PACKETS:

Column	Type	Description & Values	
S_TIME	1D	Time [s] when the space packet is sent to the telemetry: data recorder (DR) or ground station; calculated and filled by SIRIUS	
R_TIME	1D	Time [s] when the space packet is received by the ground station. If the space packet is recorded on DR in orbit and reprocessed when the spacecraft is in communication with a ground station, R_TIME is not the same as S_TIME; recorded on ground and filled by SIRIUS.	
L32TI	1J	Packet TI Lower 32 bits	2^-6 s
PROC_STATUS	32X	Record bad telemetry or bad values	
ETIEXT	1K	Extended part of the ETI (extended TI)	0 to 9999999999 (10^10 - 1)
ETIBASE	1E	Base part of the ETI (extended TI)	0 to 9999999999999999999999 (10^22-1)
UTC	20A	UTC time corresponding to S_TIME	String: yyyymmddhhmmssxxxxxx where xxxxxx represents six bits of fractional seconds.
RATE	1E	Clock rate	Hz
FLAG_ERR	1L	Flag indicating error (1) or normal (0). When a discontinuity is found in TI, the FLAG_ERR becomes error (1)	0 or 1
PASSID	10A	Identifier of the operation pass, in the format of YYMMDD9999	
U40TI	1K	Packet TI Upper 40 bits	2^-6 s
L20TI	1J	Packet TI Lower 20 bits	2^-6 s
UTC_ERT	20A	UTC in YYYYMMDDhhmmssffffff, where ffffff is in the unit of microsec. Time when the time packet received at the ground station.	String: yyyymmddhhmmssxxxxxx where xxxxxx represents six bits of fractional seconds.
ANTENNA	8A	Antenna name	ISEXT, USC34, USC20
BITRATE	1J	Bitrate in bps	999999999
BAND	1A	Transponder band	S or X
ANPFNAME	14A	File name of the antenna operation time, which is used to derive the propagation delay time from the spacecraft to the ground station	YYMMDD9999.anp
ANPTIME	14A	Date when the antenna operation time file is generated in YYYYMMDDhhmmss format.	
DLRANGE	1E	Downlink range in km, distance from the antenna to the spacecraft,	0.000 to 9999999999999.999 km

CODETYPE	12A	Coding type of the telemetry	UNCHECKED
DELAYBIT	1E	Delay bit of the TIME telemetry, in bit, signed.	-32.000000
DELAYTIM	1E	Fixed delay time of the TIME telemetry, in second.	0.000000

TIM_LOOKUP:

Column	Type	Description & Values	
TIME	1D	Mission elapsed time	Time in seconds since 01 Jan 2014 00:00:00
L32TI	1D	Packet TI Lower 32 bits	2^-6 s
STATUS	10X	Status	

2) HXI
a) HXI/event_cl

Files are:
ah[OBS_ID]hx[1|2]_[p|s][0-9]camrec_cl.evt.gz
ah[OBS_ID]hx[1|2]_[p|s][0-9]camrecpse_cl.evt.gz

“camrec” is normal (science) events
“camrecpse” is pseudo-events, used for deadtime correction.

Both science and pseudo-events have the following extensions:

Extension	Description
EVENTS	Cleaned events
GTI	Good Time Interval for cleaned events

Both science and pseudo-events have the same columns in the event extension:

Column	Description & Values
TIME	Mission elapsed time Time in seconds since 01 Jan 2014 00:00:00
OCCURRENCE_ID	Sequential Number for occurrence
CATEGORY	Data recorder priority High =84 Medium=100 Low=116
FLAG_SEU	Single-event upset flag 1-4 Si 5 CdTE
FLAG_LCHK	Length Check flag 1-4 Si 5 CdTE
FLAG_TRIG	Trigger origin TRG8: Trigger from calibration mode, TRG7: Pseudo trigger, TRG6: Forced trigger, TRG5: Trigger from CdTe-DSD layer, TRG4: 4th DSSD layer, TRG3: 3rd DSSD layer, TRG2: 2nd DSSD layer, TRG1: 1st DSSD layer
FLAG_TRIGPAT	Trigger pattern TRG8: Trigger from calibration mode, TRG7: Pseudo trigger, TRG6: Forced trigger, TRG5: Trigger from CdTe-DSD layer, TRG4: 4th DSSD layer, TRG3: 3rd DSSD layer, TRG2: 2nd DSSD layer, TRG1: 1st DSSD layer

FLAG_HITPAT	BGO shield hit pattern Always 0 in cleaned events
FLAG_FASTBGO	Fast BGO shield hit pattern Always 0 in cleaned events
LIVETIME	Time since previous occurrence 24-bit counter, unit
PROC_STATUS	Record bad telemetry or bad values Always 0 in cleaned events
STATUS	Occurrence status Flags b00000000 ok b10000000 if bad or noisy b01000000 if PHA out of min and max (no EPI overflows because is real) b00100000 PHA-Comm_noise <0 b00010000 PHA-Comm_noise >0 but < intervalX(1) or > intervalX(last) b00001000 proc_status !== 0 b00000100 length check flag b00000010 alternative gain
EVTCAT	Event category of reconstructed event Event category of reconstructed event. 1 = absorption, 2-5 fluorescence (2 = CdTe-1st DSSD 3 = CdTe-2nd DSSD, 4 = CdTe-3rd DSSD 5 = CdTe-4th DSSD)
SIGNAL	Number of Signals per side 0,1 or 2 for cleaned events; (index = 0 1 2 3 4 Top layer index =5 6 7 8 9 bottom layer)
SIGPOS	Signal position per side (X or Y of the larger EPI of the candidate event) 0:127; (index = 0 1 2 3 4 Top layer index =5 6 7 8 9 bottom layer)
SIGEPI	Sum of EPI per side (index = 0 1 2 3 4 Top layer index =5 6 7 8 9 bottom layer)
GOODBAD	Flag to indicate signal quality per side 0,1,2,3,4; (index = 0 1 2 3 4 Top layer index =5 6 7 8 9 bottom layer); 0=good signal, 1= null signals, 2=signals below threshold, 3=no signals, 4=signals null and below threshold
VALIDHITS	Hit validity per layer 0 = no hits; 1=valid hits; 2= invalid hits; (index = 0 1 2 3 4 for layers)
LAYER	Layer number 0: for 1st top DSSD, 1: 2nd DSSD, 2: 3rd DSSD, 3: 4th DSSD, 4: for CdTe
ENE_TOTAL	Sum of EPI for occurrence
PI	Pulse Invariant; 0:2047
RECO_STATUS	Reconstruction Status Always 0 in cleaned events
RAWX	Event RAW X coordinate; 1:128
RAWY	Event RAW Y coordinate; 1:128
EPITOP	EPI total top layers
EPIBOT	EPI total bottom layers
EPICUT	Energy cut test code for layer marker to indicate if included=0 exclude top=1 energy too high exclude bottom=2 energy is too low No signal=3 below threshold or null=4
ACTX	Event ACTX corrected for EOB motion; 1:256
ACTY	Event ACTY corrected for EOB motion; 1:256
DETX	Event DET X coordinate; 1:256
DETY	Event DET Y coordinate; 1:256
FOCX	FOC X coordinate in common Hitomi frame; 1:2430
FOCY	FOC Y coordinate in common Hitomi frame; 1:2430
X	X coordinate in SKY frame; 1:2430
Y	Y coordinate in SKY frame; 1:2430

Keywords:

Keyword	**Description & Values**
DETNAM	Detector subsystem; Always ‘CAMERA’ for HXI
DATAMODE	Data acquisition mode; CAMERA_NORMAL1 for event files.

PSEUDOHZ	Input rate of Pseudo events (Hz). This is used for deadtime correction.

b) HXI/event_uf

ah[OBS_ID]hx[1|2].att.gz
ah[OBS_ID]hx[1|2]_cms.fits.gz
ah[OBS_ID]cm[1|2].a[0-9]_uf.fits.gz
ah[OBS_ID]hx[1|2]_[p|s][0-9]cam_uf.evt.gz
ah[OBS_ID]hx[1|2]_a[0-9]camexp_ufa.evt.gz
ah[OBS_ID]hx[1|2]_[p|s][0-9]camrec_ufa.evt.gz
ah[OBS_ID]hx[1|2]_tel.gti.gz
ah[OBS_ID]hx[1|2]_a[0-9]camfitam_ufa.evt.gz

"att" is the delta-attitude file used for correcting for EOB motion
"cms" is the CAMS offset file used in the response generation
"cm[1|2].a[0-9]_uf" is the CAMS FFF files
"cam_uf" is the original, unreconstructed event file
"camexp_ufa" is the expanded event file containing one row for each signal in one occurance of the unreconstructed event file.
"camrec_ufa" is the reconstructed event file. The columns are identical to the cleaned event file.
"tel.gti" is a GTI file indicating the times when the HXI telemetry is not saturated.
"camfitam_ufa" is the reconstructed event file for the Am241 calibration events

ah[OBS_ID]hx[1|2].att.gz. These file are created by the tool cams2att and are the "delta-attitude" file used to correct the HXI pointing for EOB motion. Since the correction factors depend on the physical location of an HXI unit with respect to the CAMS, there are two versions of this file, one for each HXI unit.

Column	**Description & Values**
TIME	Mission elapsed time; Time in seconds since 01 Jan 2014 00:00:00
QPARAM	Attitude quaternion
POINTING	Spacecraft pointing; Three vector: RA, 0:360; declination, -90:90; Roll, 0:360
EULER	Spacecraft pointing Z-Y-Z Euler angles ; Three vector: E1= RA, 0:360; E2 = 90-dec, E3=90-roll, 0:360
ACTX	Optical axis position in ACT coordinates; 0:256
ACTY	Optical axis position in ACT coordinates; 0:256
ACTX_FLOAT	Optical axis position in ACT coordinates; 0:256
ACTY_FLOAT	Optical axis position in ACT coordinates; 0:256
RAWX	Optical axis position in RAW coordinates; 1:128
RAWY	Optical axis position in RAW coordinates; 1:128
RAWX_FLOAT	Optical axis position in RAW coordinates; 1:128
RAWY_FLOAT	Optical axis position in RAW coordinates; 1:128

Keywords:

Keyword	**Description & Values**	
CAM1DATA	Data from CAMS1 included	YES or NO
CAM2DATA	Data from CAMS2 included	YES or NO
ORIGSYS	Originating coordinate system for coordinate correction	Normally RAW
DESTSYS	Destination coordinate system for coordinate correction	Normally ACT

ah[OBS_ID]hx[1|2]_cms.fits.gz. These file are created by the tool *cams2att*.

Column	**Description & Values**
TIME	Mission elapsed time; Time in seconds since 01 Jan 2014 00:00:00
DELTARAWX	X Offset to correct HXI RAW data
DELTARAWY	Y Offset to correct HXI RAW data

COSANGLE	Cosine of rotation angle to correct HXi RAW data; 0:1
SINANGLE	Cosine of rotation angle to correct HXi RAW data; -1:1
X1	X direction motion in CAMS1 system
Y1	Y direction motion in CAMS1 system
X2	X direction motion in CAMS2 system
Y2	Y direction motion in CAMS2 system
JUMPX1	Difference between successive values of X1
JUMPY1	Difference between successive values of Y1
JUMPX2	Difference between successive values of X2
JUMPY2	Difference between successive values of Y2
QUALITY1	Quality from telemetry for CAMS1
QUALITY2	Quality from telemetry for CAMS2
XDISTANCE	X distance between CAMS units (X2-X1)
YDISTANCE	X distance between CAMS units (Y2-Y1)
DELTASATX	X Offset in SAT coordinate system
DELTASATY	Y Offset in SAT coordinate system
BAD_UNITS	Which CAMS unit has bad data for row; 0,1,2
CALC_QUALITY	Calculation quality bit flags; 0:4

Keywords:

Keyword	**Description & Values**
CAM1DATA	Data from CAMS1 included; YES or NO
CAM2DATA	Data from CAMS2 included; YES or NO
ORIGSYS	Originating coordinate system for coordinate correction; Normally RAW
DESTSYS	Destination coordinate system for coordinate correction; Normally ACT

ah[OBS_ID]hx[1|2]_[p|s][0-9]cam_uf.evt.gz. The length of the HXI data records stored onboard and telemetered are variable and depend on the number of ASICS (1-40) with data and the number of readout channels within each ASIC with data (1-32). Therefore these FITS files contain many variable-length array columns. All columns are filled in the pre-pipeline, save for the final column, EPI, which is filled by the *hxisgdpha* task.

Column	**Description & Values**
TIME	Mission elapsed time; Time in seconds since 01 Jan 2014 00:00:00
S_TIME	Time for CCSDS packet stamped by SIRIUS
ADU_CNT	ADU sequence packet counter
L32TI	Packet TI Lower 32 bits, 2^-6 s
OCCURRENCE_ID	Sequential Number for occurrence
LOCAL_TIME	Local Time to calculate TIME
CATEGORY	Data recorder priority; High =84 Medium=100 Low=116
FLAGS	Collection of all Flags
FLAG_SEU	Single-event upset flag; 1-4 Si 5 CdTE
FLAG_LCHK	Length Check flag; 1-4 Si 5 CdTE
FLAG_TRIG	Trigger origin TRG8: Trigger from calibration mode, TRG7: Pseudo trigger, TRG6: Forced trigger, TRG5: Trigger from CdTe-DSD layer, TRG4: 4th DSSD layer, TRG3: 3rd DSSD layer, TRG2: 2nd DSSD layer, TRG1: 1st DSSD layer
FLAG_TRIGPAT	Trigger pattern TRG8: Trigger from calibration mode, TRG7: Pseudo trigger, TRG6: Forced trigger, TRG5: Trigger from CdTe-DSD layer,

	TRG4: 4th DSSD layer, TRG3: 3rd DSSD layer, TRG2: 2nd DSSD layer, TRG1: 1st DSSD layer
FLAG_HITPAT	BGO shield hit pattern
FLAG_FASTBGO	Fast BGO shield hit pattern
LIVETIME	Time since previous occurrence; 24-bit counter, unit
NUM_ASIC	Number of ASICs used by occurrence; 0:40
RAW_ASIC_DATA	Occurrence telemetry array; Contains all signals collected in each ASIC for the given occurrence
PROC_STATUS	Record bad telemetry or bad values; Always 0 in cleaned events
STATUS	Occurrence status Flags b00000000 ok b10000000 if bad or noisy b01000000 if PHA out of min and max (no EPI overflows because is real) b00100000 PHA-Comm_noise <0 b00010000 PHA-Comm_noise >0 but < intervalX(1) or > intervalX(last) b00001000 proc_status !== 0 b00000100 length check flag b00000010 alternative gain
ASIC_ID	Original ASIC ID; List of ASICs involved in the occurrence
ASIC_ID_RMAP	Remapped ASIC ID; 1:40
ASIC_CHIP	ASIC data flags; 1=Data in ASIC 0=no data
ASIC_TRIG	ASIC trigger flags; 1=Trigger in ASIC 0=no trigger
ASIC_SEU	ASIC single-event upset flags ; 1=Error in ASIC 0=no error
READOUT_FLAG	Readout active flag; Original flag to indicate which channel is active e.g. ADC is present.
NUM_READOUT	The number of active channels for each ASIC; 1-32
ASIC_REF	ASIC reference channel; 10-bit ADC value for the reference channel, which is used to determine the noise level in each each
ASIC_CMN	ASIC common mode noise
READOUT_ASIC_ID	Original ASIC ID for readout
READOUT_ID	Original readout ID within an ASIC; 0:31
READOUT_ID_RMAP	Readout ID remapped to a contiguous list; 1:1280
PHA	Pulse height amplitude
EPI	PHA in keV

ah[OBS_ID]hx[1|2]_a[0-9]camexp_ufa.evt.gz: Expanded event files containing the reconstruction information for each occurrence by detector side. These files are used for calibration and not useful for science analysis.

Column	**Description & Values**
TIME	Mission elapsed time; Time in seconds since 01 Jan 2014 00:00:00
OCCURRENCE_ID	Sequential Number for occurrence
CATEGORY	Data recorder priority; High =84 Medium=100 Low=116
FLAG_SEU	Single-event upset flag; 1-4 Si 5 CdTE
FLAG_LCHK	Length Check flag; 1-4 Si 5 CdTE
FLAG_TRIG	Trigger origin TRG8: Trigger from calibration mode, TRG7: Pseudo trigger, TRG6: Forced trigger, TRG5: Trigger from CdTe-DSD layer, TRG4: 4th DSSD layer, TRG3: 3rd DSSD layer, TRG2: 2nd DSSD layer, TRG1: 1st DSSD layer

FLAG_TRIGPAT	Trigger pattern TRG8: Trigger from calibration mode, TRG7: Pseudo trigger, TRG6: Forced trigger, TRG5: Trigger from CdTe-DSD layer, TRG4: 4th DSSD layer, TRG3: 3rd DSSD layer, TRG2: 2nd DSSD layer, TRG1: 1st DSSD layer
FLAG_HITPAT	BGO shield hit pattern
FLAG_FASTBGO	Fast BGO shield hit pattern
LIVETIME	Time since previous occurrence; 24-bit counter, unit
PROC_STATUS	Record bad telemetry or bad values
STATUS	Occurrence status Flags b00000000 ok b10000000 if bad or noisy b01000000 if PHA out of min and max (no EPI overflows because is real) b00100000 PHA-Comm_noise <0 b00010000 PHA-Comm_noise >0 but < intervalX(1) or > intervalX(last) b00001000 proc_status !== 0 b00000100 length check flag b00000010 alternative gain
READOUT_ID_INDEX	Readout ID remapped to a contiguous list; 1:1280
LAYER	Layer number; 0: for 1st top DSSD, 1: 2nd DSSD, 2: 3rd DSSD, 3: 4th DSSD, 4: for CdTe
PI	Pulse Invariant; 0:2047
RECO_STATUS	Reconstruction Status
RAWX	Event RAW X coordinate; 1:128
RAWY	Event RAW Y coordinate; 1:128

ah[OBS_ID]hx[1|2]_[p|s][0-9]camrec_ufa.evt.gz: Columns identical to the cleaned event files.

ah[OBS_ID]hx[1|2]_tel.gti.gz: GTI file and the columns in these files are only TSTART and TSTOP.

ah[OBS_ID]hx[1|2]_a[0-9]camfitam_ufa.evt.gz: the Am241 calibration event file. This file is created by hxievtid.

Column	**Type**	**Description**	**Values**
TIME	1D	Mission elapsed time	Time in seconds since 01 Jan 2014 00:00:00
OCCURRENCE_ID	1J	Sequential Number for occurrence	
ADU_CNT	1B	ADU sequence packet counter	
CATEGORY	1B	Data recorder priority	High =84 Medium=100 Low=116
FLAG_SEU	5X	Single-event upset flag	1-4 Si 5 CdTE
FLAG_LCHK	5X	Length Check flag	1-4 Si 5 CdTE
FLAG_TRIG	8X	Trigger origin	TRG8: Trigger from calibration mode, TRG7: Pseudo trigger, TRG6: Forced trigger, TRG5: Trigger from CdTe-DSD layer, TRG4: 4th DSSD layer, TRG3: 3rd DSSD layer, TRG2: 2nd DSSD layer, TRG1: 1st DSSD layer
FLAG_TRIGPAT	8X	Trigger pattern	TRG8: Trigger from calibration mode, TRG7: Pseudo trigger, TRG6: Forced trigger, TRG5: Trigger from CdTe-DSD layer, TRG4: 4th DSSD layer,

			TRG3: 3rd DSSD layer, TRG2: 2nd DSSD layer, TRG1: 1st DSSD layer
FLAG_HITPAT	8X	BGO shield hit pattern	
FLAG_FASTBGO	8X	Fast BGO shield hit pattern	
LIVETIME	1J	Time since previous occurrence	24-bit counter, unit [Need clarification for time increment]
PROC_STATUS	32X	Record bad telemetry or bad values	
STATUS	8X	Occurrence status Flags	b00000000 ok b10000000 if bad or noisy b01000000 if PHA out of min and max (no EPI overflows because is real) b00100000 PHA-Comm_noise <0 b00010000 PHA-Comm_noise >0 but < intervalX(1) or > intervalX(last) b00001000 proc_status !== 0 b00000100 length check flag b00000010 alternative gain
LAYER	1B	Layer number	0: for 1st top DSSD, 1: 2nd DSSD, 2: 3rd DSSD, 3: 4th DSSD, 4: for CdTe
SIDE	1B	Side number	0:9 -- 0:1 for 1st top DSSD, 2:3 2nd DSSD, 4:5 3rd DSSD, 6:7 4th DSSD, 8:9 for CdTe
ENE_TOTAL	E	Sum of EPI for occurrence	
PI	I	Pulse Invariant	0:2047
EVTCAT	1B	Event category of reconstructed event	Event category of reconstructed event. 1 = absorption, 2-5 fluorescence (2 = CdTe-1st DSSD 3 = CdTe-2nd DSSD, 4 = CdTe-3rd DSSD 5 = CdTe-4th DSSD)
RAWX	1I	Event RAW X coordinate	1:128
RAWY	1I	Event RAW Y coordinate	1:128
RECO_STATUS	16X	Reconstruction Status	

c) HXI/hk
ah[OBS_ID]hx[1|2]_a0_hk.gz: House keeping files.

3) SGD
a) SGD/event_cl

ah[OBS_ID]sg[1|2]_[p|s][0-9]cc[1-3]rec_cl.evt.gz
ah[OBS_ID]sg[1|2]_[p|s][0-9]cc[1-3]recpse_cl.evt.gz

1, 2, or 3 for CC1, CC2 or CC2 (matches DETNAM keyword)
"rec" is normal (science) events
"recpse" is pseudo-events, used for deadtime correction.
Both science and pseudo-events have the following extensions:

Extension	**Description**
EVENTS	Cleaned events
GTI	Good Time Interval for cleaned events

Both science and pseudo-events have the same columns:

Column	**Description & Values**
TIME	Mission elapsed time; Time in seconds since 01 Jan 2014 00:00:00
OCCURRENCE_ID	Sequential Number for occurrence
CATEGORY	Data recorder priority; High =85 Medium=101 Low=117
FLAG_LCHKMIO	MIO received data; 0=ok 1=error
FLAG_CCBUSY	Compton Camera busy; 1=CC busy 0=CC not busy

FLAG_HITPAT_CC	Compton Camera hit pattern b1000000 for CC1 b0100000 for CC2 b0010000 for CC3
FLAG_HITPAT	BGO shield hit pattern; Always 0 in cleaned events
FLAG_FASTBGO	Fast BGO shield hit pattern; Always 0 in cleaned events
FLAG_SEU	SEU flag; Always 0 in cleaned events
FLAG_LCHK	Length check flag; Always 0 in cleaned events
FLAG_CALMODE	Calibration mode flag; 1=calibration mode 0=other
FLAG_TRIGPAT	Trigger pattern 0-27: corresponds to 28 trays # 28=Forced, 29=Psuedo, 30=Calibration-pulse triggers. # >=32: there are more than two simultaneous triggers initiate the occurrence.
FLAG_TRIG	Trigger origin 0-27: corresponds to 28 trays # 28=Forced, 29=Psuedo, 30=Calibration-pulse triggers. # >=32: there are more than two simultaneous triggers initiate the occurrence.
LIVETIME	Time since previous occurrence; 24-bit counter, unit
PROC_STATUS	Record bad telemetry or bad values; Always 0 in cleaned events
STATUS	Occurrence status Flags b00000000 ok b10000000 if bad or noisy b01000000 if PHA out of min and max (no EPI overflows because is real) b00100000 PHA-Comm_noise <0 b00010000 PHA-Comm_noise >0 but < intervalX(1) or > intervalX(last) b00001000 proc_status !== 0 b00000100 length check flag b00000010 alternative gain
PI	Pulse Invariant; 0:2047
ENE_TOTAL	Sum of EPI for occurrence
NUMSIGNAL	Number of signals in occurrence
NUMHITS	Hit distribution; 1bits =1hits 2bits=2hits 3bits=3hits 4bits=4hits 5bits=escape
SEQ_HITS	Sequence of hits from CALDB; Values correspond to CALDB file ah_sgd_probseq_xxx.fits
DELCOMPTON	Value of DeltaG (for >2 hits); Error on difference between geometric and kinematic scattering angles
COMPTON_TH	Kinematic Compton scattering angle (θ); 0:180
COMPTON_PH	Kinematic Compton scattering angle (ϕ); -180:180
DISTANCE0	(mm) Distance between 1st two hits
OFFAXIS	Angle (degrees) between line of sight and Compton scattering angle
CAMERAX	1st hit coord camerax; -39:39
CAMERAY	1st hit coord cameray; -39:39
CAMERAZ	1st hit coord cameraz; -77:3
LIKELIHOOD	Likelihood of event; 0:1
RECO_STATUS	Reconstruction Status; Always 0 in cleaned events
MATTYPE	Material where the event originated: 1 =Si Layer, 2 =CdTe layer, 3 = multiple layers.

Keywords:

Keyword	Description & Values
DETNAM	Detector subsystem; CC1, CC2 or CC3
DATAMODE	Data acquisition mode; CAMERA_NORMAL1 for event files.
PSEUDOHZ	Input rate of Pseudo events (Hz). This is used for deadtime correction.

ah[OBS_ID]sg[1|2]_[p|s][0-9]cc[1-3]-trace.gz

Output of the task *sgdevtid* when the parameter outtracefile is set to yes. In this case, the task outputs a file containing the reconstruction information for each occurrence by detector side. The format of this file is:

b) SGD/event_uf

ah[OBS_ID]sg[1|2]_[p|s][0-9]cc[1-3]_uf.evt.gz
ah[OBS_ID]sg[1|2]_a[0-9]cc[1-3]exp_ufa.evt.gz
ah[OBS_ID]sg[1|2]_[p|s][0-9]cc[1-3]rec_ufa.evt.gz
ah[OBS_ID]sg[1|2]_tel.gti.gz

"uf" is the original, unreconstructed event file
"exp_ufa" is the expanded event file containing one row for each signal in the unreconstructed event file.
"rec_ufa" is the reconstructed event file. The columns are identical to the cleaned event file.
"tel.gti" is a GTI file indicating the times when the HXI telemetry is not saturated.

ah[OBS_ID]sg[1|2]_[p|s][0-9]cc[1-3]_uf.evt.gz. The length of the SGD data records stored onboard and telemetered are variable and depend on the number of ASICS (1-208) with data and the number of readout channels within each ASIC with data (1-64). Therefore these FITS files contain many variable-length array columns. All columns are filled in the pre-pipeline, save for the final column, EPI, which is filled by the *hxisgdpha* task.

Column	**Description & Values**
TIME	Mission elapsed time; Time in seconds since 01 Jan 2014 00:00:00
S_TIME	Time for CCSDS packet stamped by SIRIUS
ADU_CNT	ADU sequence packet counter
L32TI	Packet TI Lower 32 bits, 2^-6 s
OCCURRENCE_ID	Sequential Number for occurrence
LOCAL_TIME	Local Time to calculate TIME
CATEGORY	Data recorder priority; High =85 Medium=101 Low=117
FLAGS	Collection of all Flags
FLAG_LCHKMIO	MIO received data; 0=ok 1=error
FLAG_CCBUSY	Compton Camera busy; 1=CC busy 0=CC not busy
FLAG_HITPAT_CC	Compton Camera hit pattern b1000000 for CC1 b0100000 for CC2 b0010000 for CC3
FLAG_HITPAT	BGO shield hit pattern
FLAG_FASTBGO	Fast BGO shield hit pattern
FLAG_SEU	SEU flag
FLAG_LCHK	Length check flag
FLAG_CALMODE	Calibration mode flag; 1=calibration mode 0=other
FLAG_TRIGPAT	Trigger pattern; 0-27: corresponds to 28 trays # 28=Forced, 29=Pseudo, 30=Calibration-pulse triggers. # >=32: there are more than two simultaneous triggers initiate the occurrence.
FLAG_TRIG	Trigger origin; 0-27: corresponds to 28 trays # 28=Forced, 29=Pseudo, 30=Calibration-pulse triggers. # >=32: there are more than two simultaneous triggers initiate the occurrence.
LIVETIME	Time since previous occurrence; 24-bit counter, unit
NUM_ASIC	Number of ASICs used by occurrence; 0:40
RAW_ASIC_DATA	Occurrence telemetry array; Contains all signals collected in each ASIC for the given occurrence
PROC_STATUS	Record bad telemetry or bad values; Always 0 in cleaned events

STATUS	Occurrence status Flags; b00000000 ok b10000000 if bad or noisy b01000000 if PHA out of min and max (no EPI overflows because is real) b00100000 PHA-Comm_noise <0 b00010000 PHA-Comm_noise >0 but < intervalX(1) or > intervalX(last) b00001000 proc_status !== 0 b00000100 length check flag b00000010 alternative gain
ASIC_ID	Original ASIC ID; List of ASICs involved in the occurrence
ASIC_ID_RMAP	Remapped ASIC ID; 1:208
ASIC_CHIP	ASIC data flags; 1=Data in ASIC 0=no data
ASIC_TRIG	ASIC trigger flags; 1=Trigger in ASIC 0=no trigger
ASIC_SEU	ASIC single-event upset flags; 1=Error in ASIC 0=no error
READOUT_FLAG	Readout active flag; Original flag to indicate which channel is active e.g. ADC is present.
NUM_READOUT	The number of active channels for each ASIC; 1-32
ASIC_REF	ASIC reference channel; 10-bit ADC value for the reference channel, which is used to determine the noise level in each each
ASIC_CMN	ASIC common mode noise
READOUT_ASIC_ID	Original ASIC ID for readout
READOUT_ID	Original readout ID within an ASIC; 0:31
READOUT_ID_RMAP	Readout ID remapped to a contiguous list; 1:13312
PHA	Pulse height amplitude
EPI	PHA in keV

ah[OBS_ID]sg[1|2]_a[0-9]cc[1-3]exp_uf.evt.gz:

Column	**Description & Values**
TIME	Mission elapsed time; Time in seconds since 01 Jan 2014 00:00:00
OCCURRENCE_ID	Sequential Number for occurrence
CATEGORY	Data recorder priority; High =85 Medium=101 Low=117
FLAG_SEU	SEU flag
FLAG_LCHK	Length check flag
FLAG_TRIG	Trigger origin 0-27: corresponds to 28 trays # 28=Forced, 29=Psuedo, 30=Calibration-pulse triggers. # >=32: there are more than two simultaneous triggers initiate the occurrence.
FLAG_TRIGPAT	Trigger pattern 0-27: corresponds to 28 trays # 28=Forced, 29=Psuedo, 30=Calibration-pulse triggers. # >=32: there are more than two simultaneous triggers initiate the occurrence.
FLAG_HITPAT	BGO shield hit pattern
FLAG_FASTBGO	Fast BGO shield hit pattern
LIVETIME	Time since previous occurrence; 24-bit counter, unit
PROC_STATUS	Record bad telemetry or bad values
STATUS	Occurrence status Flags b00000000 ok b10000000 if bad or noisy b01000000 if PHA out of min and max (no EPI overflows because is real) b00100000 PHA-Comm_noise <0 b00010000 PHA-Comm_noise >0 but < intervalX(1) or > intervalX(last) b00001000 proc_status !== 0 b00000100 length check flag b00000010 alternative gain
FLAG_LCHKMIO	MIO received data; 0=ok 1=error
FLAG_CCBUSY	Compton Camera busy;1=CC busy 0=CC not busy
FLAG_HITPAT_CC	Compton Camera hit pattern

	b1000000 for CC1 b0100000 for CC2 b0010000 for CC3
FLAG_CALMODE	Calibration mode flag; 1=calibration mode 0=other
READOUT_ID_INDEX	Readout ID remapped to a contiguous list; 1:13312
PI	Pulse Invariant; 0:2047
RECO_STATUS	Reconstruction Status
MATTYPE	Material where the event originated: 1 =Si Layer, 2 =CdTe layer, 3 = multiple layers.

Contents of SGD trace file (output of sgdevtid, using parameter 'outtracefile'). This is a file containing the reconstruction information for each occurrence by detector side. It is used only for diagnostics.

Column	Type	Description & Values
TIME	1D	Mission elapsed time Time in seconds since 01 Jan 2014 00:00:00
OCCURRENCE_ID	1J	Sequential Number for occurrence
NUMSIGNAL	1I	Number of signals in occurrence
M	1I	Number of hits is occurrence
NUMPERM	1I	Number of permutations
ESCAPE_FLAG	1X	Escape energy computed 1=yes 0=no
SIGARRAY0	1PI	Signal IDs before merging
SIGARRAY1_0	1PI	Merged signals step 1_0
SIGARRAY1A_1A	1PI	Merged signals step 1A_1A
SIGARRAY1A_1B	1PI	Merged signals step 1A_1B
SIGARRAY1A_2	1PI	Merged signals step 1A_2
SIGARRAY1A_3	1PI	Merged signals step 1A_3
HITARRAY	4I	Hit IDs
EPIARRAY	4E	Hit energies
DELTAEPIARRAY	4E	Hit energy errors
E	1PE	Error in cumulative hit energies
DELTA_E	1PE	Error in E
F	1PE	F for Compton energy test
DELTA_F	1PE	Error in F
CHECK_F	96X	F test per hit 1=pass 0=fail
TEST_F	24X	F test 1=pass 0=fail
C_THETA_G	1PE	Cos theta_G for kinematic test
D_C_THETA_G	1PE	Error in cos theta_G
C_THETA_K	1PE	Cos theta_K for kinematic test
D_C_THETA_K	1PE	Error in cos theta_K
G	1PE	G for kinematic test
DELTA_G	1PE	Error in G
CHECK_G	96X	G test per hit 1=pass 0=fail
TEST_G	24X	G test 1=pass 0=fail
PROB	1PE	Sequence probability per permutation
TEST_PROB	24X	Sequence probability test 1=pass 0=fail
FOM	1PE	Tie-breaking Figure of Merit per permutation
C_THETA_G_0	1PE	Cos theta_g(0) per permutation
D_C_THETA_G_0	1PE	Error in cos theta_g(0)
C_THETA_K_0	1PE	Cos theta_k(0) per permutation
D_C_THETA_K_0	1PE	Error in cos theta_k(0)
G_0	1PE	Cos theta_g(0) - cos theta_k(0)
DELTA_G_0	1PE	Error in G_0
CLSTRSHAPE	1I	Cluster shape code in step 1_0
MERGE1_0	1X	Merging in step 1_0 1=yes 0=no

MERGE1A_1A	1X	Merging in step 1a_1a 1=yes 0=no
MERGE1A_1B	1X	Merging in step 1a_1b 1=yes 0=no
MERGE1A_2	1X	Merging in step 1a_2 1=yes 0=no
MERGE1A_3	1X	Merging in step 1a_3 1=yes 0=no
RAND1_0	1X	Max signal chosen randomly 1=yes 0=no
RAND1A_3	1X	Max signal chosen randomly 1=yes 0=no
RAND_FOM	1X	Max FOM chosen randomly 1=yes 0=no

ah[OBS_ID]sg[1|2]_[p|s][0-9]cc[1-3]rec_ufa.evt.gz: Columns identical to the cleaned event files.

ah[OBS_ID]sgd_tel.gti.gz: GTI file and the columns in these files are only TSTART and TSTOP.

c) SGD/event_hk
ah[OBS_ID]sg[1|2]_a0.hk.gz: House keeping files.

4) SXI
a) SXI/event_cl
ah[OBS_ID]sxi_[p|s][0-9][DATACLAS]_cl.evt.gz

DATACLAS is 8-digit data class (same as DATACLAS keyword).

DATACLAS is decoded to

version			
DataMode	=	0:CCD, 1:CCD12, 2:CCD34	(DETNAM keyword)
Window	=	0:off, 1:on	(DATAMODE keyword)
Burst	=	0:off, 1:on	(DATAMODE keyword)
BrightMode	=	0:off, 1:on	
CI	=	0:off, 1:on	(CISTATUS keyword)
OtherSetting			

The data file has the following extensions:

Extension	**Description**
EVENTS	Cleaned events
GTI	Good Time Interval for cleaned events

The data file EVENTS extension has the following columns:

Column	**Description & Values**
TIME	Mission elapsed time ; Time in seconds since 01 Jan 2014 00:00:00
S_TIME5X5	Time for 5x5 packet stamped by SIRIUS
S_TIME	Time for 3x3 packet stamped by SIRIUS
ADU_CNT	ADU Sequence Packet Counter for 3x3; 0:255
ADU_CNT5X5	ADU Sequence Packet Counter for 5x5; 0:255
CATEGORY	Data Recorder Priority for 3x3; High =83 Medium=99 Low=115
CATEGORY5X5	Data Recorder Priority for 5x5; High =83 Medium=99 Low=115
L32TI	Packet TI Lower 32 bits, 2^-6 s, for 3x3
L32TI5X5	Packet TI Lower 32 bits, 2^-6 s, for 5x5
SEQSTARTTIME	Sequence Start Time
CCD_ID	CCD_ID; 0: CCD1, 1: CCD2, 2: CCD3, 3: CCD4
CCD_NAME	CCD Name; CCD1, CCD2, CCD3 or CCD4
SEGMENT	Segment ID; 0: AB, 1: CD
EVENTNUMBER	Event Number from 3x3 Packet; 1:27 for cleaned events
EVENTNUMBER5X5	Event Number from 5x5 Packet; 1:27 for cleaned events
READNODE	Readout Node; 0: A or D, 1: B or C
ADCAVE	ADC Setting; 0: averaged, 1: P or R, 2: Q or S
RAWX	Event RAW X coordinate; 0:319

RAWY	Event RAW Y coordinate; 0:639
PHAS_INNER3X3	Pulse Height Amplitudes of 3x3 Pixels
P_OUTER_MOST	Outer 5x5 Pixels Hit Pattern
SUM_OUTER_MOST	Outer 5x5 Pulse Height Sum below Split Threshold
PHAS_OUTER5X5	Pulse Height Amplitudes of outer 5x5 Pixels
ACTX	Event ACTX by CCD; 1:640
ACTY	Event ACTY by CCD; 1:640
DETX	Event DET X for array; 1:1810
DETY	Event DET Y for array; 1:1810
FOCX	FOC X coordinate in common Hitomi frame; 1:2430
FOCY	FOC Y coordinate in common Hitomi frame; 1:2430
X	X coordinate in SKY frame; 1:2430
Y	Y coordinate in SKY frame; 1:2430
PHAS	Pulse Height Amplitudes of 3x3 Pixels; -32767:32767
PHAS_MASK	Flag for 3x3 Pixels; 0: Good, 1: Bad
PHASALL	Pulse Height Amplitudes of All 5x5 Pixels; -32767:32767
PHA	Pulse Height Amplitude Sum of 3x3 Pixels; 0:4095
PI	Pulse Height Invariant; 0:4095
GRADE	Grade Value for Pixel Hit Pattern; 0:6
STATUS	Event Flag; See definition below
PROC_STATUS	Record Bad Telemetry or Bad Values; Always 0 in cleaned events

STATUS column is defined explained in Appendix E.

Keyword	**Description & Values**
WINOPT	Window option; 1=on or 0=off
WIN_ST	[pixel] Window start ACT; 461=on or 1=off or
WIN_SIZE	[pixel] Window size ACT; 80=on or 640=off
CISTATUS	Charge inject ; 1=on or 0=off
CIOFFSET	Offset from the 1st (delta ACTY);80
CIPERIOD	Change inject spacing; 80
CIFIRST	First injected row ACTY; 461=window on or 1=window off
HUCLEGTH	Length of horizontal underclock region; One value per segment
VUCHEGHT	Height of vertical underclock region; One value per segment
IMGHEGHT	Height of imaging area; One value per segment; typically 640
CCDSIZE	[Pixel] Size of 1 ccd; 640
TIMTRANB	Transfer time before exposure [s]
TIMTRANA	Transfer time after exposure [s]
EXPDEADB	Deadtime before exposure [s]
EXPDEADA	Deadtime after exposure [s]
FLUSHIMB	Flush out time [s]
LASTDEAD	Last Deadtime after exposure [s]
LASTDEL	Last Integration time in exposure [s]
NOMEXPO	Period of seq start time [s]
EVENTTHR	Event threshold [adu]; One value per segment
SPTHIN	Split inner threshold [adu]; One value per segment
SPTHOUT	Split outer threshold [adu]; One value per segment
ACTVNODE	Active node; One value per segment; 1=active, 0=inactive
DARKLOW	Dark lower threshold [adu] ; One value per segment
DARKUPP	Dark upper threshold [adu]; One value per segment
HOTPIXTH	Hot pixel threshold [adu]; One value per segment
HOCSUMSK	HOC sum threshold [adu]; One value per segment
IFOFFSET	Ifoffset [adu]; One value per segment
CnSmARON	Area discriminator CCD n, Segment m; 0=off or 1=on
CnSmARIN	Area discriminator CCD n, Segment m in area;
CnSmAROU	Area discriminator CCD n, Segment m off area;

b) SXI/event_uf

ah[OBS_ID]sxi_[p|s][0-9][DATACLAS]_uf.evt.gz
ah[OBS_ID]sxi_a[0-9][DATACLAS].fpix.gz
ah[OBS_ID]sxi_a[0-9][DATACLAS].hpix.gz
ah[OBS_ID]sxi_a[0-9]exp.fits.gz
ah[OBS_ID]sxi_mode.gti.gz
ah[OBS_ID]sxi_seg.gti.gz
ah[OBS_ID]sxi_tel.gti.gz

"uf" is the reconstructed event file. **The columns are identical to the cleaned event file.**
"fpix" is the flickering pixel file.
"hpix" is the hot pixel file.
"exp" is the exposure map.
"mode" contains a GTI for each mode (DATACLAS) in the observation
"seg" contains a GTI for each segment in the observation, where SEG0=CCD0,segment0; SEG1=CCD0,segment1; SEG2=CCD1,segment0; SEG3=CCD1,segment1; SEG4=CCD2,segment0; SEG5=CCD2,segment1; SEG6=CCD3,segment0; SEG7=CCD3,segment1
"tel.gti" is a GTI file indicating the times when the SXI telemetry is not saturated.

ah[OBS_ID]sxi_a[0-9][DATACLAS].fpix.gz. These files are created by the searchflickpix tool.
The data file has the following extensions:

Extension	Description
EVENTS	Cleaned events
PIXELS	

The EVENTS extension has the same columns as the cleaned event file.

The PIXEL extension has the following columns:

Column	Description & Values
START	Time in seconds since 01 Jan 2014 00:00:00
STOP	Time in seconds since 01 Jan 2014 00:00:00
CCD_ID	CCD_ID; 0: CCD1, 1: CCD2, 2: CCD3, 3: CCD4
ACTX	Event ACTX by CCD; 1:640
ACTY	Event ACTY by CCD; 1:640
COUNTS	Detected counts in the flickering pixel
PI	Pulse Height Invariant; 0:4095
DETX	Event DET X for array; 1:1810
DETY	Event DET Y for array; 1:1810
FOCX	FOC X coordinate in common Hitomi frame; 1:2430
FOCY	FOC Y coordinate in common Hitomi frame; 1:2430
X	X coordinate in SKY frame; 1:2430
Y	Y coordinate in SKY frame; 1:2430

ah[OBS_ID]sxi_a[0-9][DATACLAS].hpix.gz. These file are generated in the pre-pipeline. There is a single extension named HOTPIX. This extension has the following columns:

Column	Description & Values
TIME	Mission elapsed time ; Time in seconds since 01 Jan 2014 00:00:00
S_TIME	Time for 3x3 packet stamped by SIRIUS
L32TI	Packet TI Lower 32 bits, 2^-6 s, for 3x3
SEQSTARTTIME	Sequence Start Time
CCD_ID	CCD_ID; 0: CCD1, 1: CCD2, 2: CCD3, 3: CCD4
SEGMENT	Segment ID; 0: AB, 1: CD
READNODE	Readout Node; 0: A or D, 1: B or C

RAWX	Event RAW X coordinate; 0:319
RAWY	Event RAW Y coordinate; 0:639
ACTX	Event ACTX by CCD; 1:640
ACTY	Event ACTY by CCD; 1:640
DETX	Event DET X for array; 1:1810
DETY	Event DET Y for array; 1:1810
YEXTEND	Length of Hot Pixels in RAWY
PROC_STATUS	Record Bad Telemetry or Bad Values
FOCX	FOC X coordinate in common Hitomi frame; 1:2430
FOCY	FOC Y coordinate in common Hitomi frame; 1:2430
X	X coordinate in SKY frame; 1:2430
Y	Y coordinate in SKY frame; 1:2430

ah[OBS_ID]sxi_a[0-9]exp.fits.gz. These files are generated in the pre-pipeline. There is a single extension named EXPOSURE. This extension has the following columns:

Column	**Description & Values**
TIME	Mission elapsed time ; Time in seconds since 01 Jan 2014 00:00:00
S_TIME	Time for 3x3 packet stamped by SIRIUS
ADU_CNT	ADU Sequence Packet Counter for 3x3; 0:255
CATEGORY	Data Recorder Priority for 3x3; High =83 Medium=99 Low=115
L32TI	Packet TI Lower 32 bits, 2^-6 s, for 3x3
SEQSTARTTIME	Sequence Start Time
CCD_ID	CCD_ID; 0: CCD1, 1: CCD2, 2: CCD3, 3: CCD4
SEGMENT	Segment ID; 0: AB, 1: CD
EVENTNUMBER	Event Number from Exposure Packet
REJOVERULD	Number of events rejected PH[0] > EVTH_UPPER
REJUNDERLLD	Number of events rejected PH[0] < EVTH_LOWER
REJARDISC	Number of events rejected by area discrimination
REJSURDISC	Number of events rejected by surround filter
REJLOCALDISC	Number of events rejected by 3x3 local max filter
COMPFLAG	Digital Electronics editing completion status ; 0=incomplete, 1=complete
CCDPRIORITY	DE processing priority; 0:1st 1:2nd 2:3rd 3:4th
SEGPRIORITY	DE processing priority; 0:1st, 1:2nd
DETECTSEG	DE processing priority; 0:1st, 1:2nd
ENA3X3PROC	DE processing priority; 0:1st, 1:2nd
ENA5X5PROC	DE processing priority; 0:1st, 1:2nd
ADINENA	Area discriminator inclusion; 0:enabled, 1:disabled
ADOU0ENA	Area discriminator exclusion 0; 0:enabled, 1:disabled
ADOU1ENA	Area discriminator exclusion 1 ; 0:enabled, 1:disabled
ADOU2ENA	Area discriminator exclusion 2; 0:enabled, 1:disabled
ADOU3ENA	Area discriminator exclusion 3; 0:enabled, 1:disabled
SURDISCENA	Surround filter; 0:enabled, 1:disabled
LOCALDISCENA	3x3 local maximum filter; 0:enabled, 1:disabled
X0_ADIN	RAWX start of area discriminator inclusion region; 0:319
X1_ADIN	RAWX stop of area discriminator inclusion region; 0:319
Y0_ADIN	RAWY start of area discriminator inclusion region; 0:639
Y1_ADIN	RAWY stop of area discriminator inclusion region; 0:639
X0_0_ADOUT	RAWX start of area discriminator exclusion region 0; 0:319
X1_0_ADOUT	RAWX stop of area discriminator exclusion region 0; 0:319
Y0_0_ADOUT	RAWY start of area discriminator exclusion region 0; 0:639
Y1_0_ADOUT	RAWY stop of area discriminator exclusion region 0; 0:639
X0_1_ADOUT	RAWX start of area discriminator exclusion region 1; 0:319
X1_1_ADOUT	RAWX stop of area discriminator exclusion region 1; 0:319
Y0_1_ADOUT	RAWY start of area discriminator exclusion region 1; 0:639
Y1_1_ADOUT	RAWY stop of area discriminator exclusion region 1; 0:639
X0_2_ADOUT	RAWX start of area discriminator exclusion region 2; 0:319

X1_2_ADOUT	RAWX stop of area discriminator exclusion region 2; 0:319
Y0_2_ADOUT	RAWY start of area discriminator exclusion region 2; 0:639
Y1_2_ADOUT	RAWY stop of area discriminator exclusion region 2; 0:639
X0_3_ADOUT	RAWX start of area discriminator exclusion region 3; 0:319
X1_3_ADOUT	RAWX stop of area discriminator exclusion region 3; 0:319
Y0_3_ADOUT	RAWY start of area discriminator exclusion region 3; 0:639
Y1_3_ADOUT	RAWY stop of area discriminator exclusion region 3; 0:639
SURTH	Threshold for surround filter
NPIX_SURTH	Number of pixels above SURTH to trigger filter
EVTH_LOWER	Lower threshold for DE event candidate
EVTH_UPPER	Upper threshold for DE event candidate
OUTER_SPLIT_TH	Threshold of outer 5x5 pixels
UCODE_ID	Microcode identifier
TRANSLINELENGTH	Horizontal line length in frame
IMGLINELENGTH	Horizontal line length in imaging area
HOCLINELENGTH	Horizontal overclock length
HUCLINELENGTH	Horizontal underclock length
IMAGEHEIGHT	Vertical line length in imaging area
VOCHEIGHT	Vertical overclock length
VUCHEIGHT	Vertical underclock length
TRANSFERDIR	Transfer direction; bit0;Rdnode: bit1 (0:A/D 1:B/C)
ADC_ID	Identity of ASIC ADC
ADC_CHAN	Identity of ADC channel
DATACLASS	DataClass identifier
DUPDATE	Status of dark update
DUPDATESTARTTIME	Start time of dark update
LLDEVTCAND	Pixel Processing Electronics lower-level discriminator threshold
ULDEVTCAND	PE upper-level discriminator threshold
ULDPIXNUM	Number of pixels PH>ULDEVTCAND
LDPIXNUM	Number of pixels LLD<=PH<=ULD
IFRAME_OFFSET	PE offset PH value
HOCSUMNUM	Number of pixels in a row of Horizontal Over Clock
NUMEVTCAND	Number of PE event candidates
LENEVTCAND	Number of DE event candidates
NUMHOTPIX	Number of PE hot pixels
LENHOTPIX	Number of DE hot pixels
SANITY	Sanity
FRAMENUM	Frame Number
FRAMETYPE	Frame Type; 0:EVT, 1:rFRM, 2:iFRM, 3:dFRM
DETNAM	CCD:all, CCD12: CCD1+2, CCD34: CCD3+4
DATAMODE	WINDOW1, WINDOW1BURST, WINDOW2, WINDOW2BURST
PROC_STATUS	Record Bad Telemetry or Bad Values

ah[OBS_ID]sxi_tel.gti.gz: GTI file and the columns in these files are only TSTART and TSTOP.

c) SXI/hk
ah[OBS_ID]sxi_a0.hk.gz: House keeping files.

5) SXS
a) SXS/event_cl

ah[OBS_ID]sxs_[p|s][0-9]px0000_cl.evt.gz

The data file has the following extensions:

Extension	**Description**

EVENTS	Cleaned events
GTI	Good Time Interval for cleaned events

The data file has the following columns:

Column	**Description & Values**
TIME	Mission elapsed time ; Time in seconds since 01 Jan 2014 00:00:00
TRIGTIME	Trigger Time of the pulse; Time in seconds since 01 Jan 2014 00:00:00
S_TIME	Time for CCSDS packet stamped by SIRIUS; Time in seconds since 01 Jan 2014 00:00:00
L32TI	Packet TI Lower 32 bits, 2^-6 s
CATEGORY	Data recorder priority; High =82 Medium=98 Low=114
ADU_CNT	ADU sequence packet counter; 0:255
PSP_ID	PSP ID; 0=PSP-A0 1=A1 2=B0 3=B1
FORMAT_VER	Packet format version
WFRB_WRITE_LP	Lap & pointer to WFRB for writing ADC sample WFRB position where the FPGA is recording the ADC sample and derivative when the CPU edits the event packet ; (reference time for the whole packet); present lap (6 bit) + pointer (18 bit) of the WFRB.
WFRB_SAMPLE_CNT	Corresponding to top of WFRB free-run counter of the SAMPLE_CNT, corresponding to the recording ADC sample & derivative
NUM_ELEM	Number of elements in the event packet; 0:255
SUM_LOST_CNT	Sum of lost counts in the event packet
ITYPE	Event grade number; 0:High-resolution primary; ; 1: Medium-resolution primary; 2: Medium-resolution secondary ; 3: Low-resolution primary ; 4: Low-resolution secondary ; 5: Baseline events; 6: Lost events ; 7: Rejected events
TYPE	Event grade string; Hp, Mp, Ms, Lp, Ls, BL, EL, Rj according to ITYPE
IPIX	Pixel number in each PSP; 0:17
PIXEL	Unique pixel number; 0:35
TRIG_LP	Lap & pointer to WFRB for the triggered event ; when the event triggers ; NOTE: This stored two meaning a) the trigger time for event or BL or ; b) the start time of the lost event interval LOST_EVT_START_LP ; Note b) is calculated as usual and the ITYPE flag that is time associated to the start time of the lost event. Set for event or baseline or Lost ; lap (6 bit) + pointer (18 bit)
QUICK_DOUBLE	Flag set to 1 for double pulse
SLOPE_DIFFER	Flag set to 1 for invalid pulse shape
LO_RES_PH	Pulse height calculated as for low-resolution events; -8192:16383
DERIV_MAX	Maximum value of the derivative; -32768:32767
RISE_TIME	Rise-time in unit of 1/4 tick (20 us nominal); 0:127
TICK_SHIFT	Number of shift in unit of tick to find the PHA; -8:7
TIME_VERNIER	Vernier to define the finest time division; 0:15
PHA	Pulse height amplitude; -32768:65535
FLAGS	Combination of event flags; 0x20*SLOPE_DIFFER + 0x10*QUICK_DOUBLE + ITYPE(0-7) ; # FLAGS QUICKDOUBLE*32 + SLOPE_DIFFER*16 + ITYPE
EL_LOST_CNT	Number of the lost events for TYPE=EL; Filtered out for cleaned events
EL_REASON	Reason why the event TYPE=EL are lost; Filtered out for cleaned events
EL_STOP_LP	Lap & pointer of WFRB when event lost stop; Filtered out for cleaned events
PREV_INTERVAL	Interval from previous event in unit of tick
NEXT_INTERVAL	Interval to next event in unit of tick
SAMPLECNT	Used to calculate TIME
SAMPLECNTTRIG	Used to calculate TRIGTIME
ACTX	Event ACTX converted from PIXEL; 1:8
ACTY	Event ACTY converted from PIXEL; 1:8
DETX	Event DET X coordinate; 1:8
DETY	Event DET Y coordinate; 1:8
FOCX	FOC X coordinate in common Hitomi frame; 1:2430
FOCY	FOC Y coordinate in common Hitomi frame; 1:2430
X	X coordinate in SKY frame; 1:2430
Y	Y coordinate in SKY frame; 1:2430
UPI	Approximate PHA conversion to eV

EPI	PHA to eV as all events are primary
EPI2	As EPI but Ms/Ls with secondary correction
PI	Pulse Invariant after gain correction; 0:32767
INDEX	Index incrementally the events
GROUPS	Associate primary and secondary
CTMULT	Record multiplicity for electrical cross-talk
STATUS	Events Flag:recoil,elect,antico,MXS; Bit Explained in Appendix E
PROC_STATUS	Record Bad Telemetry or Bad Values; Always 0 in cleaned events
SEQ	Sequence of the associated events in a group.

Keywords:

Keyword	**Description & Values**
FILTER	Filter name used; OPEN1, OPEN2, BE, FE55, ND25, POLYIMIDE; UNDEF
GATEVALV	Gate Valve position; CLOSE or OPEN
MXSONOFF	Modulated X-ray Source; ON OFF or UNKNOWN
MXSTYPE	Which MXS is on; 1,12, 2, 34, 3,4 or none
ADRMODE	ADR mode of operation; HELIUM/CRYOFREE
DEVPTHRE	Derivative Pulse threshold
SHPTEMPL	Shape templates version
DATAMODE	Data acquisition mode; PX_NORMAL, PX_MIDRES or PX_BASELINE

b) SXS/event_uf

ah[OBS_ID]sxs_a[0-9]ac_uf.evt.gz
ah[OBS_ID]sxs_a[0-9]pxcal_uf.evt.gz
ah[OBS_ID]sxs_[p|s][0-9]pxNNNN_uf.evt.gz
ah[OBS_ID]sxs_a[0-9][ENERGREF].ghf.gz
ah[OBS_ID]sxs_el.gti.gz
ah[OBS_ID]sxs_pxNNNN_exp.gti.gz
ah[OBS_ID]sxs_mxcs.gti.gz
ah[OBS_ID]sxs_mxfn.gti.gz
ah[OBS_ID]sxs_tel.gti.gz

ENERGREF is a pneumonic for the calibration line used for the gain fit:
"crka" == Cr K-α
"cuka" == Cu K-α
"pxcal" == Pixel 12 (Mn K-α)

"pxNNNN_uf" unfiltered event file. The columns are identical to the cleaned event file.
"pxcal_uf" is an event file containing only events from the Calibration Pixel 12.
"ac_uf" is the anti-coincidence event file.
The "ghf" files are gain history files. These files are all created by sxsgain and have the same columns.
"el.gti" is the GTI for lost events. This includes times when the event are lost. There two extensions one for the antico and one for the pixel data as specified in the keyword DETNAM. There are three types of extensions: GTILOST is the time for lost event per pixel; GTIFOUND is the inverse of GTILOST per pixel and GTIFOUNDALL is the inverse of GTILOST considering all pixels.
"pxNNNN_exp.gti" Pixel-dependent GTI file
"mxcs.gti" is the GTI for the coarse MXS, which shows the times when the MXS LEDs are illuminated.
"mxfn.gti" is the GTI for the fine MXS, which shows the times of the individual MXS pulses.
"tel.gti" is a GTI file indicating the times when the HXI telemetry is not saturated.

ah[OBS_ID]sxs_a[0-9]ac_uf.evt.gz (anti-coincidence event file). This file is created by the sxsanticopi tool. There is a single EVENTS extension. This extension has the following columns:

Column	**Description & Values**
TIME	Mission elapsed time ; Time in seconds since 01 Jan 2014 00:00:00
S_TIME	Time for CCSDS packet stamped by SIRIUS; Time in seconds since 01 Jan

	2014 00:00:00
L32TI	Packet TI Lower 32 bits, 2^-6 s
CATEGORY	Data recorder priority; High =82 Medium=98 Low=114
ADU_CNT	ADU sequence packet counter; 0:255
PSP_ID	PSP ID; 0=PSP-A0 1=A1 2=B0 3=B1
FORMAT_VER	Packet format version
WFRB_WRITE_LP	lap & pointer to WFRB for writing ADC sample WFRB position where the FPGA is recording the ADC sample and derivative when the CPU edits the event packet ; (reference time for the whole packet); present lap (6 bit) + pointer (18 bit) of the WFRB.
WFRB_SAMPLE_CNT	Corresponding to top of WFRB free-run counter of the SAMPLE_CNT, corresponding to the recording ADC sample & derivative
NUM_ELEM	Number of elements in the event packet; 0:255
SUM_LOST_CNT	Sum of lost counts in the event packet
ADC_SAMPLE_PEDESTAL	ADC sample pedestal to calculate PHA; -8192:8191
AC_ITYPE	0:AC, 1:BL, 2:EL, 3:PE
AC_TYPE	AC:antico, BL:baseline, EL:event-lost, ; PE:parity error
FLG_EVENT_LOST	Lost event flag; 1=lost
FLG_BASELINE	Baseline event flag
DURATION	Pulse duration exceeding the threshold; 0:255
TRIG_LP	Lap & pointer to WFRB for the triggered event ; when the event triggers ; NOTE: This stored two meaning a) the trigger time for event or BL or ; b) the start time of the lost event interval LOST_EVT_START_LP ; Note b) is calculated as usual and the ITYPE flag that is time associated to the start time of the lost event. Set for event or baseline or Lost; lap (6 bit) + pointer (18 bit)
FLG_PARITY_ERR	Flag for the event with communication error
TRIG_LAP_LSB	LSB of the lap, for debugging
ADC_SAMPLE_MAX	Maximum value of the pulse in ADC unit; -8192:8191
EL_LOST_CNT	Number of the lost events for TYPE=EL; Filtered out for cleaned events
PHA	Pulse height amplitude; -8192:16383
PI	Pulse Invariant after gain correction; -8192:12200
SAMPLECNT	Used to calculate TIME
PROC_STATUS	Record Bad Telemetry or Bad Values

Contents of ah[OBS_ID]sxs_a[0-9][ENERGREF]000.ghf.gz (gain history file). This file is created by sxsgain. The data file has the following extensions:

Extension	**Description**
Drift_energy	Drift energy
Grid_profile	Grid profile

The Drift_energy extension reports the fitting results for each spectra in the following columns:

Column	**Description & Values**
TIME	Midpoint of the time interval over which the spectrum is collected.; Time in seconds since 01 Jan 2014 00:00:00
PIXEL	Unique pixel number; 0:35
COR_FIT	Energy correction factor from spectrum fit (ratio of fitted energy to profile).
COR_AVE	Energy correction factor from spectrum average (ratio of bin average energy to profile)
CHISQ	Reduced χ2 of fit
AVGUNBIN	Average energy of events in spectrum prior to binning
AVGBIN	Weighted spectrum average energy
AVGFIT	Average energy from fit
SHIFT	Fitted energy shift
SCALE	Fitted vertical scaling factor
BGRND	Fitted background counts
SLOPE	Fitted background slope

WIDTH	Fitted Gaussian convolution width; Based on sxsgain parameter: if 'fitwidth=no', same as broadening parameter; if 'fitwidth=yes', fitted width
TELAPSE	Seconds between first and last events in spectrum
EXPOSURE	Exposure seconds between first and last events (calculated using the GTI)
NEVENT	Number of events in fitting group
BINMESH	Array containing the count spectrum energy bins
SPECTRUM	Array containing the observed binned count spectrum
FITPROF	Array containing theoretical profile with fitted parameters applied
TEMP_FIT	Temperature derived from fitted PHA and AVGFIT columns
TEMP_AVE	Temperature derived from average PHA and; AVGBIN columns
SIGSHLIKE	Standard deviation of shift from the maximum likelihood method.
SIGWDLIKE	Standard deviation of shift from the maximum likelihood method.
SIGSHCHI2	Standard deviation of shift from the χ2 method.
SIGWDCHI2	Standard deviation of width from the χ2 method.
SHIFTS	Shift values for error functions; If 'writeerrfunc' parameter is set, the chi-squared and; likelihood calculated valued are output in the arrays SHCHI2, SHLIKE,; WDCHI2 and WDLIKE. The numbers of values output in these arrays are; specified in the 'nerrshift' and 'nerrwidth' parameters, respectively.
SHLIKE	Shift likelihood values
SHCHI2	Shift reduced χ2 values
WIDTHS	Width values for error functions
WDLIKE	Width likelihood values
WDCHI2	Width reduced χ2 values

The Grid_profile extension contains the energies and amplitudes of the theoretical profile used in the fitting procedure, including any convolution from the 'broadening' parameter. This extension has the following columns: in addition to the PHA column, there is a column containing the profile amplitude for each individual pixel.

Column	**Description & Values**
PHA	Pulse height amplitude; -32768:65535
AMPLITUDE00	Profile amplitude for pixel 0
AMPLITUDE01	Profile amplitude for pixel 1
AMPLITUDE02	Profile amplitude for pixel 2
AMPLITUDE03	Profile amplitude for pixel 3
AMPLITUDE04	Profile amplitude for pixel 4
AMPLITUDE05	Profile amplitude for pixel 5
AMPLITUDE06	Profile amplitude for pixel 6
AMPLITUDE07	Profile amplitude for pixel 7
AMPLITUDE08	Profile amplitude for pixel 8
AMPLITUDE09	Profile amplitude for pixel 9
AMPLITUDE10	Profile amplitude for pixel 10
AMPLITUDE11	Profile amplitude for pixel 11
AMPLITUDE12	Profile amplitude for pixel 12
AMPLITUDE13	Profile amplitude for pixel 13
AMPLITUDE14	Profile amplitude for pixel 14
AMPLITUDE15	Profile amplitude for pixel 15
AMPLITUDE16	Profile amplitude for pixel 16
AMPLITUDE17	Profile amplitude for pixel 17
AMPLITUDE18	Profile amplitude for pixel 18
AMPLITUDE19	Profile amplitude for pixel 19
AMPLITUDE20	Profile amplitude for pixel 20
AMPLITUDE21	Profile amplitude for pixel 21
AMPLITUDE22	Profile amplitude for pixel 22
AMPLITUDE23	Profile amplitude for pixel 23
AMPLITUDE24	Profile amplitude for pixel 24
AMPLITUDE25	Profile amplitude for pixel 25
AMPLITUDE26	Profile amplitude for pixel 26
AMPLITUDE27	Profile amplitude for pixel 27
AMPLITUDE28	Profile amplitude for pixel 28

AMPLITUDE29	Profile amplitude for pixel 29
AMPLITUDE30	Profile amplitude for pixel 30
AMPLITUDE31	Profile amplitude for pixel 31
AMPLITUDE32	Profile amplitude for pixel 32
AMPLITUDE33	Profile amplitude for pixel 33
AMPLITUDE34	Profile amplitude for pixel 34
AMPLITUDE35	Profile amplitude for pixel 35

ah[OBS_ID]sxs_mxcs.gti.gz. This file is created by the tool mxsgti and contains the following extensions:

Extension	**Description**
GTIMXSCSON1	Coarse MXS LED 1 on-time
GTIMXSCSON2	Coarse MXS LED 2 on-time
GTIMXSCSON3	Coarse MXS LED 3 on-time
GTIMXSCSON4	Coarse MXS LED 4 on-time
GTIMXSCSOFF1	Coarse MXS LED 1 off-time
GTIMXSCSOFF2	Coarse MXS LED 2 off-time
GTIMXSCSOFF3	Coarse MXS LED 3 off-time
GTIMXSCSOFF4	Coarse MXS LED 4 off-time
GTIMXSCSON13	Coarse merged MXS LED 1 and 3 on-time
GTIMXSCSON24	Coarse merged MXS LED 2 and 4 on-time
GTIMXSCSOFF13	Coarse merged MXS LED 1 and 3 off-time
GTIMXSCSOFF24	Coarse merged MXS LED 2 and 4 off-time

Keywords:

Keyword	**Description & Values**
LED1LEN	LED 1 Pulse length [s]
LED1SPC	LED 1 Pulse spacing [s]
LED2LEN	LED 2 Pulse length [s]
LED2SPC	LED 2 Pulse spacing [s]
LED3LEN	LED 3 Pulse length [s]
LED3SPC	LED 3 Pulse spacing [s]
LED4LEN	LED 4 Pulse length [s]
LED4SPC	LED 4 Pulse spacing [s]

ah[OBS_ID]sxs_mxfn.gti.gz. These files are created by the tool mxsgti and contain the following extensions:

Extension	**Description**
GTIMXSFNON1	Fine MXS LED 1 on-time
GTIMXSFNON2	Fine MXS LED 2 on-time
GTIMXSFNON3	Fine MXS LED 3 on-time
GTIMXSFNON4	Fine MXS LED 4 on-time
GTIMXSFNOFF1	Fine MXS LED 1 off-time
GTIMXSFNOFF2	Fine MXS LED 2 off-time
GTIMXSFNOFF3	Fine MXS LED 3 off-time
GTIMXSFNOFF4	Fine MXS LED 4 off-time
GTIMXSFNON13	Fine merged MXS LED 1 and 3 on-time
GTIMXSFNON24	Fine merged MXS LED 2 and 4 on-time
GTIMXSFNOFF13	Fine merged MXS LED 1 and 3 off-time
GTIMXSFNOFF24	Fine merged MXS LED 2 and 4 off-time

Keywords:

Keyword	**Description & Values**
LED1LEN	LED 1 Pulse length [s]
LED1SPC	LED 1 Pulse spacing [s]

LED2LEN	LED 2 Pulse length [s]
LED2SPC	LED 2 Pulse spacing [s]
LED3LEN	LED 3 Pulse length [s]
LED3SPC	LED 3 Pulse spacing [s]
LED4LEN	LED 4 Pulse length [s]
LED4SPC	LED 4 Pulse spacing [s]

Contents of ah[OBS_ID]sxs_xxx_pxcal.ghf.gz (gain history file). This file is created by sxsgain.
The data file has the following extensions:

Extension	**Description**
Drift_energy	Drift energy
Grid_profile	Grid profile

The Drift_energy extension reports the fitting results for each spectrum in the following columns:

Column	**Description & Values**
TIME	Midpoint of the time interval over which the spectrum is collected.; Time in seconds since 01 Jan 2014 00:00:00
PIXEL	Unique pixel number; 0:35
COR_FIT	Energy correction factor from spectrum fit (ratio of fitted energy to profile).
COR_AVE	Energy correction factor from spectrum average (ratio of bin average energy to profile)
CHISQ	Reduced $\chi 2$ of fit
AVGUNBIN	Average energy of events in spectrum prior to binning
AVGBIN	Weighted spectrum average energy
AVGFIT	Average energy from fit
SHIFT	Fitted energy shift
SCALE	Fitted vertical scaling factor
BGRND	Fitted background counts
SLOPE	Fitted background slope
WIDTH	Fitted Gaussian convolution width; Based on sxsgain parameter: if 'fitwidth=no', same as broadening parameter; if 'fitwidth=yes', fitted width
TELAPSE	Seconds between first and last events in spectrum
EXPOSURE	Exposure seconds between first and last events (calculated using the GTI)
NEVENT	Number of events in fitting group
BINMESH	Array containing the count spectrum energy bins
SPECTRUM	Array containing the observed binned count spectrum
FITPROF	Array containing theoretical profile with fitted parameters applied
TEMP_FIT	Temperature derived from fitted PHA and AVGFIT columns
TEMP_AVE	Temperature derived from average PHA and; AVGBIN columns
SIGSHLIKE	Standard deviation of shift from the maximum likelihood method.
SIGWDLIKE	Standard deviation of shift from the maximum likelihood method.
SIGSHCHI2	Standard deviation of shift from the $\chi 2$ method.
SIGWDCHI2	Standard deviation of width from the $\chi 2$ method.
SHIFTS	Shift values for error functions; If 'writeerrfunc' parameter is set, the chi-squared and; likelihood calculated valued are output in the arrays SHCHI2, SHLIKE,; WDCHI2 and WDLIKE. The numbers of values output in these arrays are; specified in the 'nerrshift' and 'nerrwidth' parameters, respectively.
SHLIKE	Shift likelihood values
SHCHI2	Shift reduced $\chi 2$ values
WIDTHS	Width values for error functions
WDLIKE	Width likelihood values
WDCHI2	Width reduced $\chi 2$ values

;
The Grid_profile extension contains the energies and amplitudes of the theoretical profile used in the fitting procedure, including any convolution from the 'broadening' parameter. This extension has the following columns: in addition to the PHA column, there is a column containing the profile amplitude for each individual pixel.; ;

Column	**Description & Values**

PHA	Pulse height amplitude; -32768:65535
AMPLITUDE00	Profile amplitude for pixel 0
AMPLITUDE01	Profile amplitude for pixel 1
AMPLITUDE02	Profile amplitude for pixel 2
AMPLITUDE03	Profile amplitude for pixel 3
AMPLITUDE04	Profile amplitude for pixel 4
AMPLITUDE05	Profile amplitude for pixel 5
AMPLITUDE06	Profile amplitude for pixel 6
AMPLITUDE07	Profile amplitude for pixel 7
AMPLITUDE08	Profile amplitude for pixel 8
AMPLITUDE09	Profile amplitude for pixel 9
AMPLITUDE10	Profile amplitude for pixel 10
AMPLITUDE11	Profile amplitude for pixel 11
AMPLITUDE12	Profile amplitude for pixel 12
AMPLITUDE13	Profile amplitude for pixel 13
AMPLITUDE14	Profile amplitude for pixel 14
AMPLITUDE15	Profile amplitude for pixel 15
AMPLITUDE16	Profile amplitude for pixel 16
AMPLITUDE17	Profile amplitude for pixel 17
AMPLITUDE18	Profile amplitude for pixel 18
AMPLITUDE19	Profile amplitude for pixel 19
AMPLITUDE20	Profile amplitude for pixel 20
AMPLITUDE21	Profile amplitude for pixel 21
AMPLITUDE22	Profile amplitude for pixel 22
AMPLITUDE23	Profile amplitude for pixel 23
AMPLITUDE24	Profile amplitude for pixel 24
AMPLITUDE25	Profile amplitude for pixel 25
AMPLITUDE26	Profile amplitude for pixel 26
AMPLITUDE27	Profile amplitude for pixel 27
AMPLITUDE28	Profile amplitude for pixel 28
AMPLITUDE29	Profile amplitude for pixel 29
AMPLITUDE30	Profile amplitude for pixel 30
AMPLITUDE31	Profile amplitude for pixel 31
AMPLITUDE32	Profile amplitude for pixel 32
AMPLITUDE33	Profile amplitude for pixel 33
AMPLITUDE34	Profile amplitude for pixel 34
AMPLITUDE35	Profile amplitude for pixel 35

ahsxs_adr.gti.gz: GTI file GTI file to extract gain-unstable times in and after ADR cycles**.**

Extension	**Description**
GTIADRON	GTI for the ADR ON
GTIADROFF	GTI for the ADR OFF

ah[OBS_ID]sxx_tel.gti.gz: GTI file and the columns in these files are only TSTART and TSTOP.

c) SXS/hk
ah[OBS_ID]sxs_a0_hk.gz: House keeping files.

APPENDICE E: SXS, SXI, HXI and SGD Status Flags

1) **SXS**

STATUS BIT	**MEANING**
STATUS[1]=1	Event is contained within a general GTI
STATUS[2]=1	Event is contained within a pixel-specific GTI
STATUS[3]=1	Event is coincident with an antico event
STATUS[4]=1	Event occurs near in time to another event
STATUS[5]=1	Event occurs near in time to another pixel 12 event (recoil cross-talk)
STATUS[6]=1	Bit 4 is set and the recoil energy test is satisfied
STATUS[7]=1	Event occurs near in time to an event in an electrically-adjacent pixel
STATUS[8]=1	Within group of electrical cross-talk events, this event has the largest PHA
STATUS[9]=1	Event is contained within the direct MXS GTI
STATUS[10]=1	Event is contained in the afterglow region of a direct MXS GTI
STATUS[11]=1	Event is contained within the indirect MXS GTI
STATUS[12]=1	Event is contained in the afterglow region of an indirect MXS GTI
STATUS[13]=1	Same as STATUS[7], but using 2nd electrical cross-talk delta-time
STATUS[14]=1	Same as STATUS[8], but using 2nd electrical cross-talk delta-time
STATUS[15]=1	Not assigned
STATUS[16]=1	Not assigned

2) **SXI**

STATUS BIT	**MEANING**
STATUS[1]=1	All bad events set by "bad_status" parameter
STATUS[2]=1	Inside the calibration source region
	OUT OF AREA:
STATUS[3]=1	Out of CCD
STATUS[4]=1	Out of window
STATUS[5]=1	Out of area discrimination
	PIXELS:
STATUS[6]=1	CI row
STATUS[7]=1	Bad pixel from CALDB
STATUS[8]=1	Bad column from CALDB
STATUS[9]=1	Hot pixel from pre-pipeline
STATUS[10]=1	Flickering pixel
	BOUNDARIES:
STATUS[11]=1	CCD boundary
STATUS[12]=1	Window boundary
STATUS[13]=1	Segment boundary
STATUS[14]=1	Area discrimination boundary
STATUS[15]=1	At least one 3x3 surrounding pixel has a bad status
	NEIGHBORS:
STATUS[16]=1	CI trailing row
STATUS[17]=1	CI preceding row
STATUS[18]=1	Preceding/following of bad column
STATUS[19]=1	Neighbors of bad/hot pixel and bad column
STATUS[20]=1	Neighbors of flickering pixel
STATUS[21]=1	Neighbors of preceding/following of bad column
STATUS[22]=1	Neighbors of CCD/window boundary
STATUS[23]=1	Neighbors of segment boundary
	OTHERS:
STATUS[24]=1	(sxiphas) 3x3 info is present but 5x5 is absent
STATUS[25]=1	(sxiphas) 3x3 is absent
STATUS[26]=1	(sxipi - general) PHAS[0] < event threshold
STATUS[27]=1	(sxipi - vtevnodd) Video temperature is out of range

STATUS[28]=1	(sxipi - vtevnodd) Lack of video temp HK at time close to the event
STATUS[29]=1	(sxipi - chtrail/CTI) Correction value is negative
STATUS[30]=1	(sxipi - general) Null value by correction process
	DIAGNOSTICS:
STATUS[31]=1	1st trailing row of the CI rows
STATUS[32]=1	1st preceding row of the CI rows
STATUS[33]=1	2nd trailing row of the CI rows
STATUS[34]=1	2nd preceding row of the CI rows
STATUS[35]=1	3rd trailing row of the CI rows
STATUS[36]=1	3rd preceding row of the CI rows
STATUS[37]=1	All bad events set by "bad_status" parameter
STATUS[38]=1	Inside the calibration source region
	OUT OF AREA:
STATUS[39]=1	Out of CCD
	NOT ASSIGNED:
STATUS[40]=1	Not assigned
STATUS[41]=1	Not assigned
STATUS[42]=1	Not assigned
STATUS[43]=1	Not assigned
STATUS[44]=1	Not assigned
STATUS[45]=1	Not assigned
STATUS[46]=1	Not assigned
STATUS[47]=1	Not assigned

3) HXI

RECO_STATUS	**MEANING**
RECO_STATUS=b0	Good event (photo-absorption or fluorescence)
RECO_STATUS=b001	STATUS is bad
RECO_STATUS=b010	FASTBGO or HITPAT is high (BGO veto signal exists)
RECO_STATUS=b100	No good event identified

4) SGD

BIT	**RECO_STATUS BIT**	**MEANING**
RECO_STATUS[1]	RECO_PSEUDO	To identify the PSEUDO data : If NUM_ASIC=0 &TRIG_PAT[29]=1 & TRIG_PAT[28] &TRIG_PAT[30]=0
RECO_STATUS[2]	RECO_CALMODE	To identify the CALMODE data : If TRIG_PAT[30]=1 & TRIG_PATH[28] &TRIG_PATH[29]=0
RECO_STATUS[3]	RECO_READALL	To identify the READALL data : If NUM_ASIC=ALIVE_ASIC & (TRIG_PAT[28]=1 or TRIG_PAT[29]=1) &TRIG_PAT[30]=0
RECO_STATUS[4]	RECO_NOT SURE	To identify any other data that do not fit status bit = 1 2 3
RECO_STATUS[5]	RECO_BAD_PROC STATUS	To identify occurrences with PROC_STATUS= bad (from FFF or Time assignment)
RECO_STATUS[6]	RECO_SKIP_RECO	If CALMODE or READALL occurrences and parameter skipreco =yes
RECO_STATUS[7]	RECO_NO_SIGNALS	If occurrence do not have signals e.g. NUM_ASIC=0
RECO_STATUS[8]	RECO_FASTBGO_HITPAT	If occurrence has FLAG_HITPAT or FLAG_FASTBGO set & parameter rejectbgo=yes
RECO_STATUS[9]	RECO_TOO_MANY_SIGNALS	If the number of signals is higher than the parameter numsignal (default =48)
RECO_STATUS[10]	RECO_ALL_SIGNALS_LOW	If all signals are below threshold
RECO_STATUS[11]	RECO_CLUSTER_TOO_MAN_SIGNAL	Step 1-0 if a cluster has more than 5 signal

	S	
RECO_STATUS[12]	RECO_CLUSTER_WRONG_SHAPE	Step 1-0 if a cluster has invalid shape
RECO_STATUS[13]	RECO_SI_SI_GROUP_TOO_LARGE	Step 1a-3 if there are more than 3 SI signals in the same electron scattering group
RECO_STATUS[14]	RECO_TOO_MANY_HITS	Step 2 check if more than 4 hits : if more can not test the sequence
RECO_STATUS[15]	RECO_ALL_BAD_F_M_EQ_2	If 2 hits : F test fails for both sequence
RECO_STATUS[16]	RECO_ALL_BAD_F_M_GT_2	If > 2 hits : F test fails for all sequences
RECO_STATUS[17]	RECO_ALL_BAD_G	If > 2 hits : G test fails for all sequences
RECO_STATUS[18]	RECO_ALL_LOW_PROB_M_EQ_2	Hits=2 & both sequences have a sequence probability too low lower than the parameter probaccept2
RECO_STATUS[19]	RECO_ALL_LOW_PROB_M_GT_2	Hits > 2 & all sequences have a sequence probability too low lower than the parameter probaccept3 or probaccept4
RECO_STATUS[20]	RECO_SINGULARITY_IN_ESCAPE_CALC	Singularity in escape energy calculation (divide by zero)
RECO_STATUS[21]	RECO_ONE_SIGNAL	Occurrence has one valid signal
RECO_STATUS[22]	RECO_ONE_GOOD_SIGNAL	Occurrence has one non-NULL signal above threshold
RECO_STATUS[23]	RECO_ONE_HIT_REMAINING_1_0	Occurrence has one valid hit after Step 1-0
RECO_STATUS[24]	RECO_ONE_HIT_REMAINING_1A_1A	Occurrence has one valid hit after Step 1a-1a
RECO_STATUS[25]	RECO_ONE_HIT_REMAINING_1A_1B	Occurrence has one valid hit after Step 1a-1b
RECO_STATUS[26]	RECO_ONE_HIT_REMAINING_1A_2	Occurrence has one valid hit after Step 1a-2
RECO_STATUS[27]	RECO_ONE_HIT_REMAINING_1A_3	Occurrence has one valid hit after Step 1a-3
RECO_STATUS[28]	RECO_ONE_SEQUENCE_REMAINING_F	Occurrence has one valid sequence after F tests
RECO_STATUS[29]	RECO_ONE_SEQUENCE_REMAINING_G	Occurrence has one valid sequence after G tests
RECO_STATUS[30]	RECO_ONE_SEQUENCE_REMAINING_PROBM	Occurrence has one sequence above parameter probaccept2/3/4
RECO_STATUS[31]	RECO_ONE_SEQUENCE_REMAINING_TIE_BREAK	Occurrence has one sequence after tie break (figure of merit)
RECO_STATUS[32]		
RECO_STATUS[33]	RECO_GOTO_ESCAPE_CALC_VIA_F_FAILURE	Perform escape energy iteration if all F tests fail
RECO_STATUS[34]	RECO_GOTO_ESCAPE_CALC_VIA_G_FAILURE	Perform escape energy iteration if all G tests fail
RECO_STATUS[35]	RECO_GOTO_ESCAPE_CALC_VIA_PROBM_FAILURE	Perform escape energy iteration if all sequence probabilities are low
RECO_STATUS[36]	RECO_PERFORM_ESCAPE_CALC	Escape energy iteration was performed

APPENDICE F: Science Pipeline

Data	**Task**	**Description**
SXS		
MXS data	mxsgti	Assign time and calculate GTI on MXS data
Antico	sxsanticopi	Calculate PI on antico data
Events	coordevt	Assign coordinates on Pixel data
Events	sxsflagpix	Flag pixel Step1
Events	sxssecid	Associate secondary events to primary /Recalculate event grades
Events	sxsgain	Calculate gain using pixel 12
Events	sxspha2pi	Calculate PI Step 1
Events	sxsflagpix	Flag pixel also for recoil Step2
Events	sxssecid	Associate secondary events to primary /Recalculate event grades
Events	sxsseccor	Correct PHA for secondary events
Events	sxspha2pi	Assign PI (Step 2)
Alternative Gain	sxsgain	if MXS is on calculate gain using CrKa, CuKa MgKa AlKa, run separately for each energy line but not often to data
Event screening	ahgtigen	Create GTI using ehk and mkf
Event screening	ahscreen	GTIPOINT, GTITEL, GTIEHK ,GTIMKF, GTIMXSFNOFF13, GTIMXSFNOFF24 & event screening
PI	TLMIN=0/ TLMAX=32767	Adjust the TLMIN/TLMAX of PI column
SXI		
Event	sximodegti	Create GTI excluding dead time for each SXI observing
Hot pixel file	coordevt	Assign coordinates on pixel data (optional)
Event	coordevt	Assign coordinates on pixel data
Event	sxiphas	Merge inner 3x3 and outer 5x5 pulse height columns in SXI event list
Event	sxiflagpix	Flag pixel STATUS for SXI event data
Event	sxipi	Calculate pulse invariant (PI) values and assign grades for SXI events
Event	sxipi	Calculate pulse invariant (PI) values and assign grades for SXI events
Event	Filter	STATUS[1]==b0
Event	searchflickpix	Search for anomalous 'flickering' pixels in event files from CCD-type detectors
Flickering pixel event	coordevt	Assign coordinates on pixel data
Event file with flickering pixels	sxiflagpix	Flag pixel STATUS for SXI event data
Event screening	ahgtigen	Create GTI using ehk and mkf
Event screening	ahscreen	GTIPOINT, GTITEL, GTI_<DATACLTM>, GTIEHK ,GTIMKF& event screening
HXI		
CAMS data	cams2att	Computes a time-dependent delta-attitude file
HXI event data	hxisgdsff,	Converts an HXI or SGD First FITS File (FFF) into the Second FITS File (SFF)
HXI event data	hxisgdpha	Calibrates the HXI or SGD PHA for each signal in the SFF event file
HXI event data	hxievtid	Reconstructs HXI events
HXI event data	coordevt	Convert events from one coordinate system to another
Event screening	ahgtigen	Create GTI using ehk and mkf
Event screening	ahscreen	GTIPOINT, GTITEL, GTIEHK , GTIMKF& event screening
Pseudo-event screening	ahgtigen	Create GTI using ehk and mkf
Pseudo-event screening	ahscreen	GTIPOINT, GTITEL, GTIEHK, GTIMKF& event screening

SGD		
SGD event data	hxisgdsff,	Converts an HXI or SGD First FITS File (FFF) into the Second FITS File (SFF)
SGD event data	hxisgdpha	Calibrates the HXI or SGD PHA for each signal in the SFF event file
SGD event data	sgdevtid	Reconstructs SGD events
Event screening	ahgtigen	Create GTI using ehk and mkf
Event screening	ahscreen	GTIPOINT, GTITEL, GTIEHK, GTIMKF& event screening
Pseudo-event screening	ahgtigen	Create GTI using ehk and mkf
Pseudo-event screening	ahscreen	GTIPOINT, GTITEL, GTIEHK, GTIMKF& event screening

APPENDICE G: List of files coming from the pipeline

In all what follows, OBS_ID is 8-digit observation ID (same as OBS_ID keyword), 1 or 2 for HXI1, SGD1 or HXI2, SGD2 (matches INSTRUME keyword), p or s for pointing or slew (matches OBS_MODE keyword), and 0-9 is file number when data is split into multiple files

OBS_ID		
Directory	**Filename**	**Content**
/auxil		**Common to all sequences**
	ahxxxxxxxxxxgen_a0.hk	General House Keeping file
	ahxxxxxxxxxx.cat	Catalog file
	ahxxxxxxxxxx.att	Attitude File
	ahxxxxxxxxxx.orb	Orbit File
	ahxxxxxxxxxx.tim	Time file
	ahxxxxxxxxxx.ehk	Extended House Keeping file
	ahxxxxxxxxxx.ehk2	Extended House Keeping file with antico rate
	ahxxxxxxxxxx.mkf	MKF file
	ahxxxxxxxxxx.com	Command file
	ahxxxxxxxxxx_gen.gti	General GTI file
	ahxxxxxxxxxx_fff.cat	Catalog file
/log		
	ahxxxxxxxxxx_job.par	Parameter file for the archive
	ahxxxxxxxxxx_lv1.par	Parameter file
	ahxxxxxxxxxx_errlog.html	Errors which occur during the processing
	ahxxxxxxxxxx_index.html	Index of the html files
	ahxxxxxxxxxx_joblog.html	General log of the processing
	ahxxxxxxxxxx_flinfo.html	List of files
	ahxxxxxxxxxx_hdpage.html	General html page
/hxi/event_uf		**HXI Unfiltered Events**
	ahxxxxxxxxxxcm1_a0_uf.fits ahxxxxxxxxxxcm2_a0_uf.fits	CAMS FFF files.
	ahxxxxxxxxxxhx1_cms.fits ahxxxxxxxxxxhx2_cms.fits	CAMS offset file used in the response generation
	ahxxxxxxxxxxhx1_cms.gti ahxxxxxxxxxxhx2_cms.gti	CAMS offset file GTI
	ahxxxxxxxxxxhx1.att ahxxxxxxxxxxhx2.att	Delta-attitude file used for correcting for EOB motion
	ahxxxxxxxxxxhx1_aRsclhst.fits ahxxxxxxxxxxhx2_aRsclhst.fits	Shield data (with two extensions: scalar and an histogram)
	ahxxxxxxxxxxhx1_aRbst.fits ahxxxxxxxxxxhx2_aRbst.fits	Shield data (burst)
	ahxxxxxxxxxxhxi_scan.gti	GTI scan for shield
	ahxxxxxxxxxxhxi_tel.gti	GTI telemetry saturation
	ahxxxxxxxxxxhx1_sRcam_uf.evt ahxxxxxxxxxxhx2_sRcam_uf.evt	Slew data. Science file FFF
	ahxxxxxxxxxxhx1_pRcam_uf.evt ahxxxxxxxxxxhx2_pRcam_uf.evt	Point data. Science file FFF
	ahxxxxxxxxxxhx1_sRcamrec_ufa.evt ahxxxxxxxxxxhx2_sRcamrec_ufa.evt	Slew data. Reconstructed CAM events
	ahxxxxxxxxxxhx1_pRcamrec_ufa.evt ahxxxxxxxxxxhx2_pRcamrec_ufa.evt	Point data. Reconstructed CAM events
	ahxxxxxxxxxxhx1_aRcamexp_ufa.evt ahxxxxxxxxxxhx2_aRcamexp_ufa.evt	Expanded event file containing one row for each signal in the unreconstructed event file.
	ahxxxxxxxxxxhx1_aRcamfitam_ufa.evt ahxxxxxxxxxxhx2_aRcamfitam_ufa.evt	
/hxi/event_cl		**HXI Cleaned events**
	ahxxxxxxxxxxhx1_pRcamrec_cl.evt ahxxxxxxxxxxhx2_pRcamrec_cl.evt	Pointing. Reconstructed events.

	ahxxxxxxxxxxhx1_pRcamrecpse_cl.evt ahxxxxxxxxxxhx2_pRcamrecpse_cl.evt	Pointing. Reconstructed pseudo-events.
/hxi/hk		**HXI House Keeping files**
	ahxxxxxxxxxxhx1_a0.hk ahxxxxxxxxxxhx2_a0.hk	House Keeping for for HXI1 and HXI2
/hxi/products		**Product files (To be populated)**
		img, lightcurve, spectrum
/sgd/event_uf		**SGD Unfiltered Events**
	ahxxxxxxxxxxsg1_aRsclhst.fits ahxxxxxxxxxxsg2_aRsclhst.fits	Shield data (with two extensions: scalar and an histogram)
	ahxxxxxxxxxxsg1_aRbst.fits ahxxxxxxxxxxsg2_aRbst.fits	Shield data (burst)
	ahxxxxxxxxxxsgd_scan.gti	GTI scan for shield.
	ahxxxxxxxxxxsgd_tel.gti	GTI for telemetry saturation
	ahxxxxxxxxxxsg1_pRccN_uf.evt ahxxxxxxxxxxsg2_pRccN_uf.evt	Pointing data. Science data. One for each SGD and each CC.
	ahxxxxxxxxxxsg1_sRccN_uf.evt ahxxxxxxxxxxsg2_sRccN_uf.evt	Slew data. Science data. One for each SGD and each CC.
	ahxxxxxxxxxxsg1_aRccNexp_ufa.evt ahxxxxxxxxxxsg2_aRccNexp_ufa.evt	Expanded event file containing one row for each signal in the unreconstructed event file. One for each SGD and each CC.
	ahxxxxxxxxxxsg1_pRccNrec_ufa.evt ahxxxxxxxxxxsg2_pRccNrec_ufa.evt	Reconstructed event file. One for each SGD and each CC. Pointing
	ahxxxxxxxxxxsg1_sRccNrec_ufa.evt ahxxxxxxxxxxsg2_sRccNrec_ufa.evt	Reconstructed event file. One for each SGD and each CC. Slew.
/sgd/event_cl		**SGD Cleaned events**
	ahxxxxxxxxxxsg1_pRccNrec_cl.evt ahxxxxxxxxxxsg2_pRccNrec_cl.evt	Pointing. Reconstructed events.
	ahxxxxxxxxxxsg1_pRccNrecpse_cl.evt ahxxxxxxxxxxsg2_pRccNrecpse_cl.evt	Pointing. Reconstructed pseudo-events.
/sgd/hk		**SGD House Keeping files**
	ahxxxxxxxxxxsg1_a0.hk ahxxxxxxxxxxsg2_a0.hk	House Keeping for SGD1 and SGD2
/sgd/products		**Product files (To be populated)**
		lightcurve, spectrum
/sxi/event_uf		**SXI Unfiltered Events**
	ahxxxxxxxxxxsxi_a0exp.fits	SXI exposure
	ahxxxxxxxxxxsxi_tel.gti	GTI telemetry saturation
	ahxxxxxxxxxxsxi_pRyyyyyyyy_uf.evt	Pointing. SXI science data
	ahxxxxxxxxxxsxi_sRyyyyyyyy_uf.evt	Slew. SXI science data
	ahxxxxxxxxxxsxi_aRyyyyyyyy.hpix	SXI bad pixels file
	ahxxxxxxxxxxsxi_mode.gti	GTI for each mode (DATACLAS) in the observation

	ahxxxxxxxxxxsxi_seg.gti	contains a GTI for each segment in the observation, where SEG0=CCD0,segment0;
		SEG1=CCD0,segment1;
		SEG2=CCD1,segment0;
		SEG3=CCD1,segment1;
		SEG4=CCD2,segment0;
		SEG5=CCD2,segment1;
		SEG6=CCD3,segment0;
		SEG7=CCD3,segment1

	ahxxxxxxxxxxsxi_pRyyyyyyyy.img	
	ahxxxxxxxxxxsxi_aRyyyyyyyy.fpix	SXI flickering pixels file
/sxi/event_cl		**SXI Cleaned events**
	ahxxxxxxxxxxsxi _pRyyyyyyyy_cl.evt	Pointing. Cleaned events
/sxi/hk		**SXI House Keeping files**
	ahxxxxxxxxxxsxi_a0.hk	House Keeping for SXI
/sxi/products		**Product files (To be populated)**
		img, lightcurve, spectrum
/sxs/event_uf		**SXS Unfiltered Events**

	ahxxxxxxxxxxsxs_tel.gti	GTI for telemetry saturation
	ahxxxxxxxxxxsxs_el.gti	GTI of lost event (2 extensions for pixel and antico), This file contains the **bad** time
	ahsxs_adr.gti	The GTI file to extract gain-unstable durations in and after ADR cycles
	ahxxxxxxxxxxsxs_mxfn.gti	GTI MXS on/off fine
	ahxxxxxxxxxxsxs_mxcs.gti	GTI MXS on/off coarse
	ahxxxxxxxxxxsxs_aRac_uf.evt	Antico slew+pointing SFF
	ahxxxxxxxxxxsxs_pRpxfxxx_uf.evt	Pixel SFF pointing
	ahxxxxxxxxxxsxs_sRpxfxxx_uf.evt	Pixel SFF Slew
	ahxxxxxxxxxxsxs_aRpxcalfxxx_uf.evt	Pixel 12 slew+pointing SFF
	ahxxxxxxxxxxsxs_xxx_pxcal.ghf	Gain file for pixel 12
	ahxxxxxxxxxxsxs_xxx_cua.ghf	Gain file for line CuKα
	ahxxxxxxxxxxsxs_xxx_cra.ghf	Gain file for line CrKα
	ahxxxxxxxxxxsxs_xxx_ala.ghf	Gain file for line AlKα
	ahxxxxxxxxxxsxs_xxx_mga.ghf	Gain file for line MgKα
	ahxxxxxxxxxxsxs_pxfxxx.gti	GTI file for all pixels
	ahxxxxxxxxxxsxs_pxfxxx_exp.gti	Pixel-dependent GTI file
/sxs/event_cl		**SXS Cleaned events**
	ahxxxxxxxxxxsxs_aRpxfxxx_cl.evt	All (pointing and slew) cleaned event file
/sxs/hk		**SXS House Keeping files**
	ahxxxxxxxxxxsxs_a0.hk1	HK1 SXS
/sxs/products		**Product files**
		img, lightcurve, spectrum

APPENDIX H: ACRONYMS (To be check LAST== when the document is finished!)

Term or Acronym	Definition
APD	Avalanche Photo Diode
ARF	Auxiliary Response File
ASCII	American Standard Code for Information Interchange
ASIC	Application Specific Integrated Circuit
BGO	Bismuth Germanate Oxide - $Bi_4Ge_3O_{12}$
BI	Back illuminated
CALDB	CALibration DataBase
CAMs	Canadian ASTRO-H Metrology System
CC	Compton Camera
CCD	Charge-Coupled Device
CdTe	Cadmium Telluride
CIAO	Chandra Interactive Analysis of Observations
COR	Cut-Off Rigidity
CTI	Charge Transfer Inefficiency
CXRB	Cosmic X-Ray background
Dec	Declination
DSD	Double-sided Strip Detectors
DSSD	Double-sided Si-Strip Detectors
EA	Effective Area
EOB	Extensible Optical Bench
EPI	Energy-proportional Pulse-height Invariant
FFF	First Fits File
FITS	Flexible Image Transport System
FoV	Field of View
FTP	File Transfer Protocol
FWHM	Full Width Half Max
GIF	Graphics Interchange Format
GPG	Gnu Privacy Guard
GSFC	Goddard Space Flight Center
GTI	Good Time Interval

HEASARC	High Energy Astrophysics Science Archive Research Center
HK	House Keeping
HP	High resolution Primary
HR/MR/LR	High/Mid/Low Resolution
HTML	Hyper Text Markup Language
HXD	Hard X-ray Detector
HXI	Hard X-ray Imager
HXT	Hard X-ray Telescope
ISAS	Institute of Space and Astronautical Science
JAXA	Japan Aerospace eXploration Agency
LP	Low-resolution Primary
LS	Low-resolution Secondary
LSF	Line-Spread Function
MET	Mission Elapsed Time
MIO	Mission Input/Output
MP	Mid-resolution Primary
MS	Mid-resolution Secondary
MXS	Modulated X-ray Source
NRA	NASA Research Announcement
NASA	National Aeronautics and Space Administration
NXB	Non X-ray Background
OGIP	Office of General Investigator Program
OS	Operating System
PDF	Portable Document File
PGP	Pretty Good Privacy
PHA	Pulse Height Amplitude
PI	Principal investigator / Pulse Invariant
PSF	Point Spread Function
QE	Quantum Efficiency
RA	Right Ascension

RMF	Redistribution Matrix File
RPT	Raw Packet Telemetry
SAA	South Atlantic Anomaly
SAO	Smithsonian Astrophysical Observatory
SFF	Second Fits File
SGD	Soft Gamma-ray Detector
SXI	Soft X-ray Imager
SXS	Soft X-ray Spectrometer
SXT	Soft X-ray Telescope
SXT-I	Soft X-ray Telescope Imager (associated with SXI)
SXT-S	Soft X-ray Telescope Spectrometer (associated with SXS)
UPI	Uncorrected Pulse Invariant
US	United States
USC	Uchinoura Space Center
UT	Universal Time
UV	Ultra Violet
WCS	World Coordinate Systems
WMAP	Weighted Map